Multi-view NRSfM: Affordable Setup for High-Fidelity 3D Reconstruction

Mosam Dabhi

CMU-RI-TR-21-12

May 7, 2021

The Robotics Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA

Thesis Committee:
Simon Lucey, *chair*
Laszlo Jeni, *co-chair*
Katerina Fragkiadaki
Nathaniel Chodosh

Submitted in partial fulfillment of the requirements for the degree of Master of Science in Robotics.

To all my advisors and mentors.

Abstract

Triangulating a point in 3D space should only require two corresponding camera projections. However in practice, expensive multi-view setups – involving tens sometimes hundreds of cameras – are required to obtain the high fidelity 3D reconstructions necessary for many modern applications. In this thesis, we argue that similar fidelity can be obtained using as little as two cameras by breaking the tenet of rigidity which is central to much of modern multi-view geometry. Our approach instead leverages recent advances in Non-Rigid Structure from Motion (NRSfM) using neural shape priors while also enforcing multi-view equivariance. We show how our method can achieve comparable fidelity to expensive multi-view rigs using only two physical camera views.

Acknowledgments

I would like to start by expressing sincere gratitude to my dear advisors, Simon Lucey, Ian Fasel, and Laszlo Jeni for taking me in as their advisee. I remember taking my Computer Vision class in the first semester of my Masters and approaching Simon with a research proposal in mind. Ever since my interview with him, I was certain that I wanted to join his research group, and feel grateful for the opportunity to have been a part of it. Simon and Ian taught me how to collect my ever wandering ideas into a potentially cool idea. I learned from them how to always think about the "Why?" of the problem first. Simon taught me how to think broad, yet not losing focus from the ultimate long-term agenda. His consistent reminder for being confident and broad is helping me shape into the researcher I aspire to become. I am thankful for their continued support and guidance as I embark on my Ph.D. journey ahead.

I am extremely grateful to my fiancée, Shraddha for being on my side, keeping me sane, and for being my core strength all these years – thank you for entertaining my numerous random research discussions! I would also like to thank my brother Meet for his unconditional love and for being a pillar of support whenever I needed it. Finally, I would like to thank my father Sanjay Dabhi. I would not have been fortunate enough to pursue research at such a prestigious university had it not been for his relentless support, love, beliefs, and amazing understanding for myself and Meet.

Contents

When this dissertation is viewed as a PDF, the page header is a link to this Table of Contents.

List of Figures

List of Tables

Chapter 1

Background

This thesis focuses on devising affordable setups for high-fidelity 3D reconstruction of objects in the wild. If we operate over rigid objects that maintain their structure over time, we can leverage many conventional Structure-from-Motion (SfM) based approaches prevalent in the Simultaneous Localization and Mapping (SLAM) scenarios. However, 3D reconstruction of objects in the wild is non-trivial, especially because most of the objects we encounter deform non-rigidly over time, such as the human body, human face, human hands, animal body, and so on.

In a restricted environment setup such as the ones in industry and academia, there are usually very sophisticated setups for high-fidelity 3D reconstruction. Multi-view setup rigs with specialized hardware for storage, gen-locking camera exposures, and ground-truth camera parameters are utilized in these sophisticated setups. On the other hand, going in-the-wild and collecting data cheaply where we don't necessarily have such a sophisticated setup brings a question: Could we make the use of this widely available cheap data to generate similar high-fidelity 3D labels. Solving the problem of non-rigid 3D reconstruction comes under the domain of Non-rigid Structure-from-Motion (NRSfM). NRSfM tries to predict the 3D keypoints directly from the 2D annotations. These 3D keypoints could act as 3D labels in a conventional deep learning-based architecture that runs the regression frameworks to train a 3D lifting pipeline, as we have shown in Fig. 1.2. The problem with the regression frameworks shown in Fig. 1.2 is that they require accurate 3D labels, and it is tough to get those high-fidelity 3D labels for supervision in an affordable way. This makes

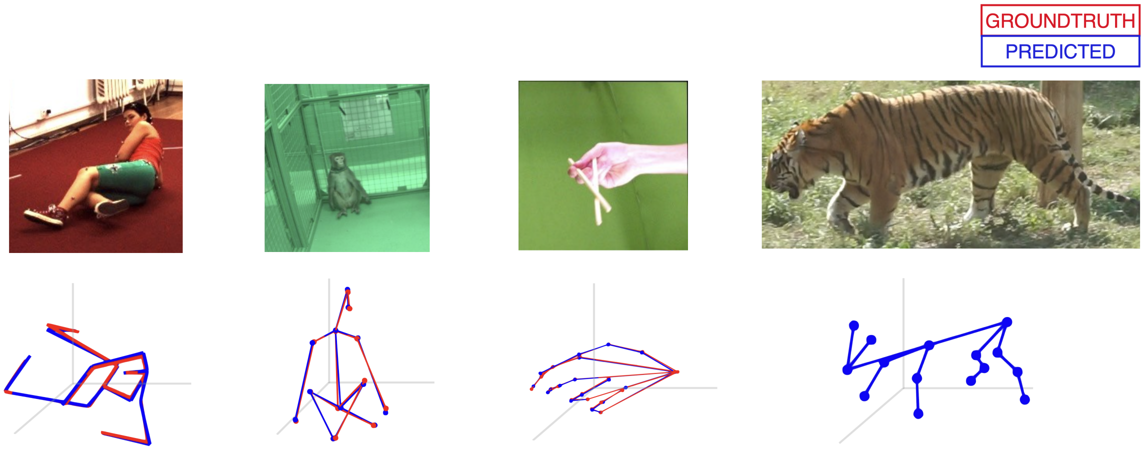

Figure 1.1: The set of 3D shapes describing different object categories (e.g. human body, monkeys, hand hands, or tiger body) is inherently nonrigid. The work proposed in this thesis discusses an affordable setup to generate high-fidelity 3D reconstructions of these non-rigid object categories as shown above. Empirically (see the 3D reconstructions in the second row), we demonstrate that our approach can achieve comparable fidelity to expensive multi-view rigs using minimal setup. Blue lines depict the predicted structure from the proposed approach and red lines show the corresponding groundtruth (if available).

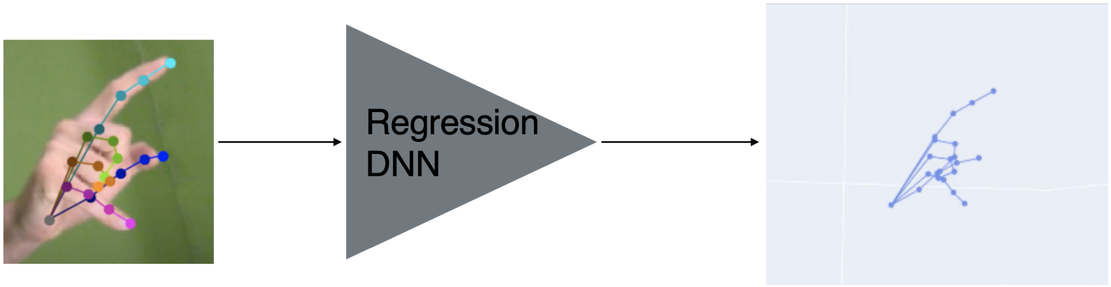

Figure 1.2: A traditional setup using regression frameworks requires accurate 3D labels for running the above pipeline.

labeling in 3D a very costly operation if we have only images in the wild. However, 2D annotations on the other hand are very easy to obtain where a human could easily annotate the 2D annotations over sets of images collected in-the-wild, as shown in Fig. 1.4

More broadly, the topics discussed in this thesis are about separating the geometry problem from the vision problem. Learning neural priors using either spatial or temporal information and applying these techniques to solve 3D problems could be

Figure 1.3: Images captured in-the-wild using widely available smartphone camera from multiple views.

immensely helpful in the quest to generate affordable setups for 3D reconstruction. If we could solve this problem from a geometric perspective before we even look at the image could have an immense potential, the first being collecting and reasoning about abundant data available in-the-wild.

As mentioned above, the 2D landmarks data could be easily annotated by humans. Traditionally, different geometrical constraints are then applied over the given 2D data to generate the 3D labels out of 2D annotations. In hindsight, we can say that it is easier to collect a much larger in-the-wild dataset of just images from multiple views captured by millions of smartphone cameras as shown in Fig. 1.3, compared to a constrained setup of industrial lab or academia. The question that this thesis tries to answer then is: *Could we apply some neural priors over the 3D structures to reconstruct them in an affordable way from multi-view 2D annotations collected in-the-wild?*

Taking inspiration from modern deep learning literature, we propose applying neural priors over the spatial structure to generate the 3D structures. We choose to impose neural shape priors through hierarchical sparsity constraints [26] literature for approaching this solution. On this note, we first discuss the background behind hierarchical sparsity prior introduced by Kong et al. [26], where each non-rigid shape is represented by a sequence of hierarchical dictionaries - commonly referred to as sparse dictionary learning problem. This chapter lays a background for the methodologies discussed in this thesis. We first discuss the notations used to describe the problem domain of Non-rigid Structure-from-Motion (NRSfM), followed by an in-depth discussion of hierarchical sparsity neural prior. Finally, the thesis layout is presented with detailed chapter descriptions.

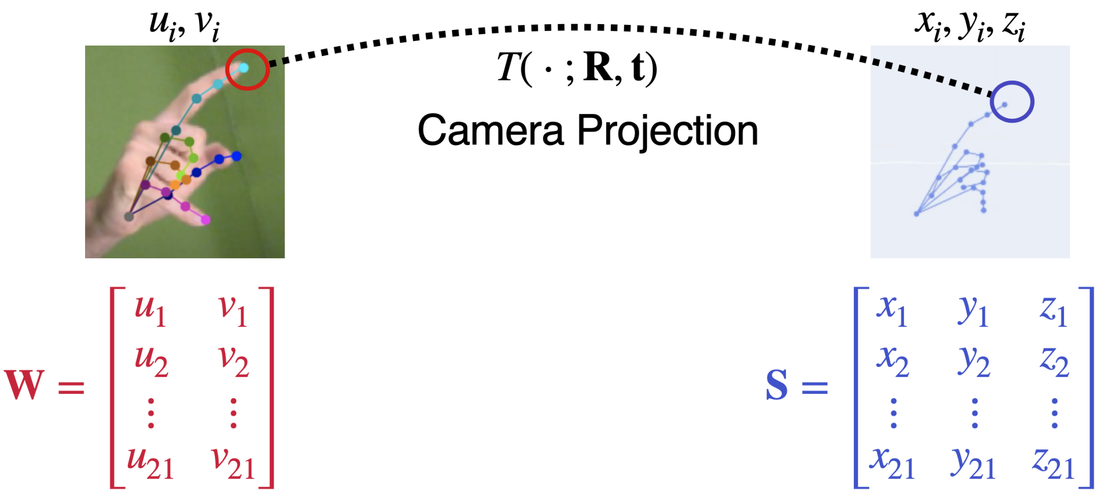
u_i, v_i
x_i, y_i, z_i
$T(\,\cdot\,;\mathbf{R},\mathbf{t})$
Camera Projection
$\mathbf{W} = \begin{bmatrix} u_1 & v_1 \\ u_2 & v_2 \\ \vdots & \vdots \\ u_{21} & v_{21} \end{bmatrix}$
$\mathbf{S} = \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ \vdots & \vdots & \vdots \\ x_{21} & y_{21} & z_{21} \end{bmatrix}$

Figure 1.4: Natural 2D annotations, $\mathbf{W}$ are the projection of rotated/translated 3D data, $\mathbf{S}$ using the rotation and translation matrix, $\mathbf{R}$ and $\mathbf{t}$, respectively.

1.1 Notations

This thesis uses the following notations throughout the manuscript.

Variable type	Examples
Scalar	s, N, K, L
Vector	$\mathbf{s}, \boldsymbol{\psi}, \boldsymbol{\lambda}$
Matrix	$\mathbf{W}, \mathbf{S}, \mathbf{R}, \mathbf{t}, \mathbf{D}$
Function	$\boldsymbol{f_e}, \boldsymbol{f_d}, \boldsymbol{g}$
l^{th} layer	${}^{l}\boldsymbol{\psi},\ {}^{l}\mathbf{D},\ {}^{l}\boldsymbol{\lambda}$
n^{th} instance	$\mathbf{W}^{(n)}, \mathbf{S}^{(n)}, \boldsymbol{\psi}^{(n)}, \boldsymbol{\lambda}^{(n)}$
k^{th} view	$\mathbf{W}_k, \mathbf{R}_k, \boldsymbol{\psi}_k$

Any different signs utilized to explain a mathematical phenomenon other than the ones described above would be explicitly defined wherever deemed necessary.

1.2 Non-rigid Structure-from-Motion (NRSfM)

This work relies on the fact that the natural 2D annotations are the projection of the transformed 3D data in space. Having said that, our task is to factor out the 3D structure, $\mathbf{S}$ and the camera matrix, $\mathbf{R}$ corresponding to the projection from the

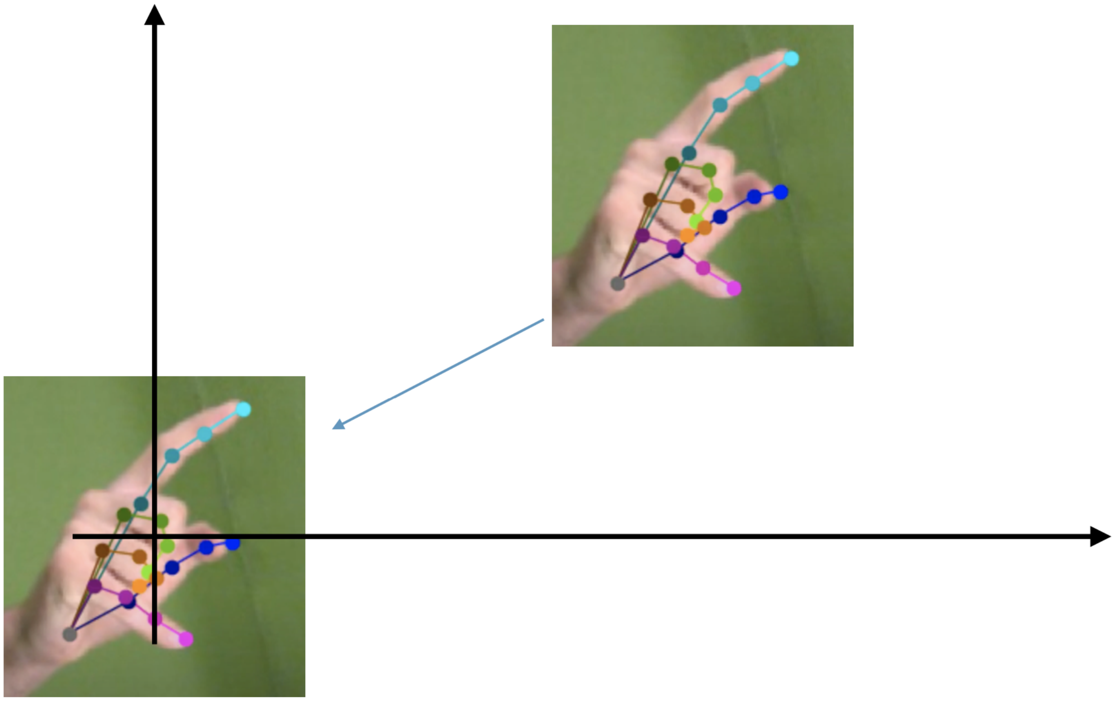

Figure 1.5: If all the points are visible, we could compute the center of the object and shift it to origin — thereby eliminating the camera translation.

given 2D annotations, $\mathbf{W}$. However, we need to enforce some neural shape prior over the structure, $\mathbf{S}$ to obtain a unique solution. For the image on the left shown in Fig. 1.4 having the 2D annotations $\mathbf{W}$, we could factor out the 3D keypoints, $\mathbf{S}$ along with the camera matrices. Since we are trying to calculate the 2D keypoint annotations, $\mathbf{W}$ the camera matrix is a 3×2 matrix for obtaining the 2D key points. We assume a weak-perspective effect, supported by the fact that the camera would be placed reasonably far enough from the object.

The 3D reconstruction problem deals with the problem of factorizing this 2D projection matrix, $\mathbf{W}$ as the product of the 3D shape matrix, $\mathbf{S}$ and the rotation matrix, $\mathbf{R}$, using just the known input pose, $\mathbf{W}$. This problem is referred to as a Structure for Motion (SfM). However, since we are dealing with a non-rigid object such as a hand, we refer to it as a Non-Rigid Structure from Motion (NRSfM). Usually, this problem is solved by decomposing $\mathbf{W}$ into $\mathbf{SR}$.

If it was just a rigid structure, then we could assume a prior on $\mathbf{S}$ that forces the shape to remain consistent across frames by enforcing *rank*-3 as it is a rigid shape [45].

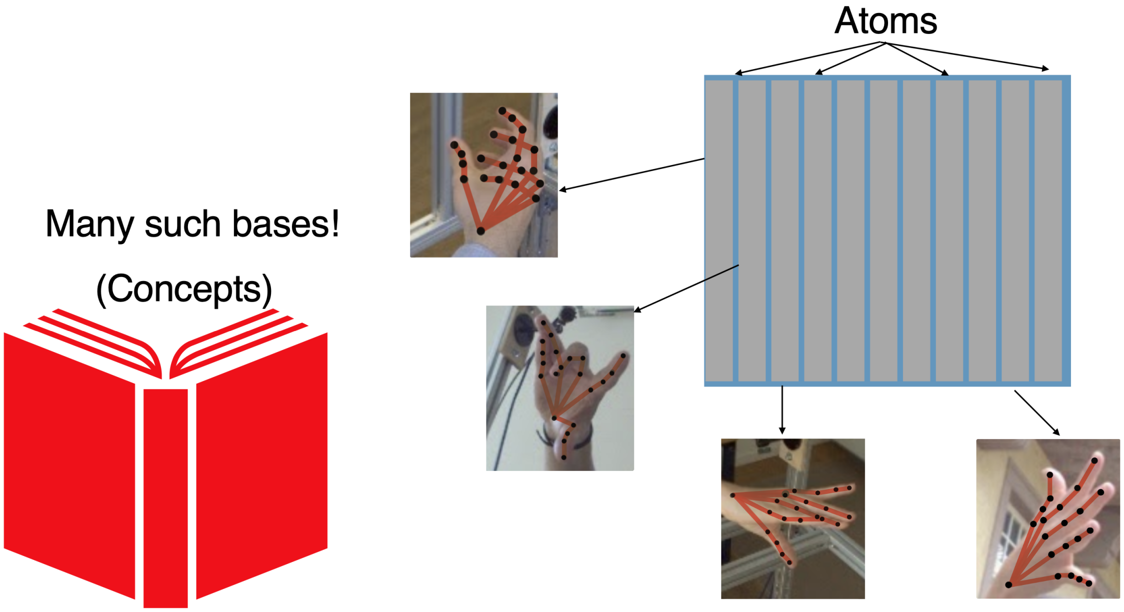

Figure 1.6: The dictionary on the left could be considered as "codebook". Each "atom" is a basic unit that can be used to "compose" larger units.

However, since we are dealing with non-rigid shapes, this prior cannot be applied since the shape is deforming across the different frames. Different priors on the shape have been applied in the literature to solve this problem such as *low-rank* [2, 6, 8, 9, 30] prior instead of *rank*-3 priors, and other such priors are discussed later in the thesis. One such neural shape prior we are specifically interested is the sparsity prior.

1.3 Sparsity prior

We plan to leverage sparse dictionary learning to solve the problem of applying neural shape priors, where we learn an overcomplete dictionary that encompasses all the non-rigid variations of the 2D projections, as shown in Fig. 1.6. Inside these dictionary columns that we call atoms, we have many such bases or concepts, shown in Fig. 1.6. The goal for dictionary learning is to have as much variation within the atoms as possible so that we could encapsulate the variations of the shapes among the non-rigid object categories. Using bases (concepts) that we know, we could learn this dictionary as shown in Fig. 1.6. Over-complete dictionaries are leveraged here because they cover the manifold of most of the data available in our non-rigid domain.

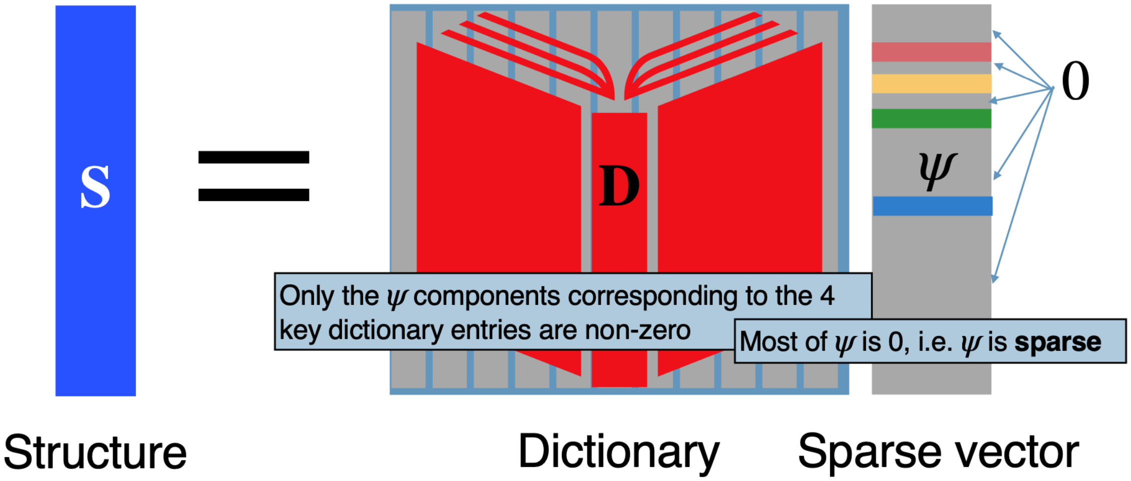
S
=
D
ψ
0
Only the ψ components corresponding to the 4 key dictionary entries are non-zero
Most of ψ is 0, i.e. ψ is **sparse**
Structure
Dictionary
Sparse vector

Figure 1.7: $\boldsymbol{\psi}$ is a sparse vector. A combination of these entries with the dictionary could represent our input with the shown setup of $\mathbf{D} \times \boldsymbol{\psi}$ that equals the 3D structure, $\mathbf{S}$.

Hence, if we can capture most of the non-rigid bases, the final 3D reconstruction representation could retrieve multiple 3D reconstruction solutions.

We learn a dictionary, $\mathbf{D}$ along with a sparse code, $\boldsymbol{\Psi}$ that acts as a sparse shape prior over the given structure, $\mathbf{S}$. In this prior, each nonrigid shape is represented by a sequence of dictionaries and corresponding non-negative sparse codes hierarchically — in a multi-layered format. Each sparse code is determined by its lower-level neighbor and affects the next level. These additional layers actually result in a more constrained and more stable sparse code recovery process. This insight breaks the combinatorial explosion of the number of subspaces and consequently maintains the robustness of sparse code recovery. Based on this observation, we are able to utilize substantially overcomplete dictionaries to model a highly deformable object from a large-scale image collection without worrying about constructability and robustness.

1.4 Nonlinearity in shape prior

This section delineates the nonlinearity that is enforced within this neural prior. Instead of assuming that the shape is a linear combination of shape basis, this approach assumes that the shape comes from a nonlinear mapping from $\boldsymbol{\psi}$ to the $\mathbf{S}$

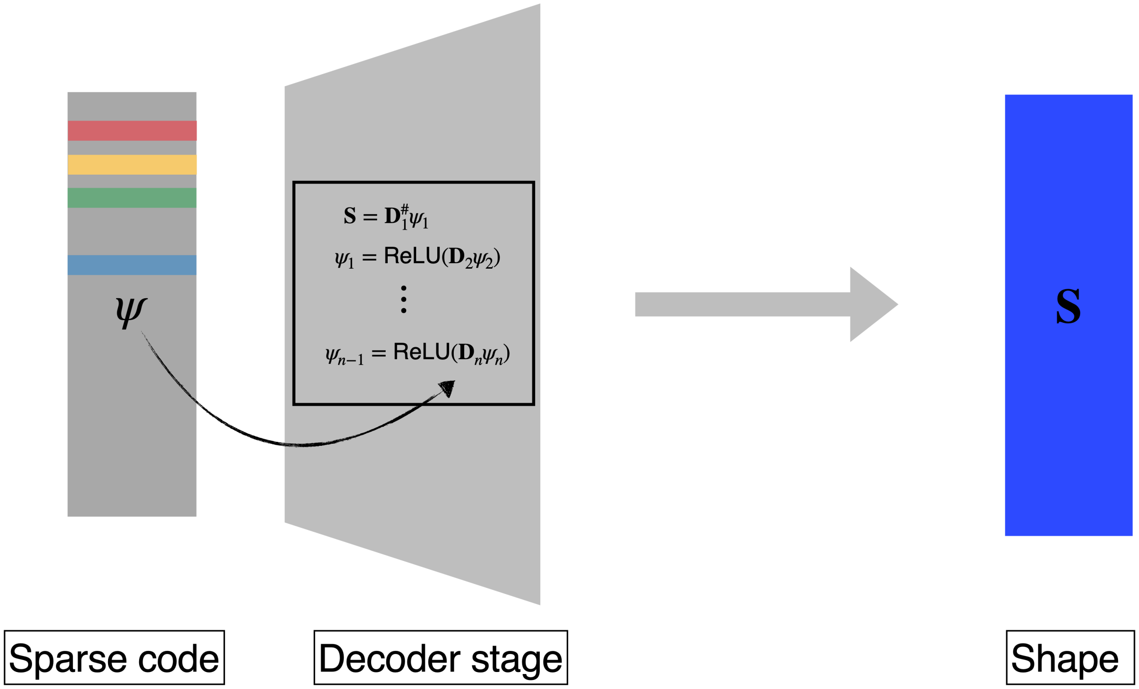
ψ
$\mathbf{S} = \mathbf{D}_1^{\#}\psi_1$
$\psi_1 = \text{ReLU}(\mathbf{D}_2\psi_2)$
$\vdots$
$\psi_{n-1} = \text{ReLU}(\mathbf{D}_n\psi_n)$
$\mathbf{S}$
Sparse code
Decoder stage
Shape

Figure 1.8: Visualization of where the nonlinearity is coming into play within the sparsity neural shape prior. The subscripts denote the hierarchical layers for the overcomplete dictionaries and sparse codes.

as shown in Fig. 1.8. Thus, the nonlinear mapping could be understood within the decoder visualization shown in Fig. 1.8.

1.5 Thesis Outline

To convey the ideas documented above, this thesis is structured as follows –

- Chapter 2 details the motivation for the setup about going in-the-wild, collecting data cheaply, and reason about them for the task of 3D reconstruction when multiple views are provided. It sets up the motivation for leveraging the proposed neural shape priors for high-fidelity multi-view 3D reconstruction.
- Chapter 3 details the work that discusses relevant literature concerning 3D reconstruction as well as 3D deep learning methodologies with different settings of available information modalities. This chapter talks about relevant approaches that has multi-view, single-view, full 3D supervision, to weak 3D supervision,

and finally talks about approaches with no 3D supervision.

- Chapter 4 sets up the base approach of this manuscript by discussing the methodologies about multi-view neural shape prior.
- Finally, Chap. 5 presents extensive evaluations of the proposed approach across numerous benchmarks and object categories including the human body, human hands, and monkey body. This chapter shows how the proposed method is able to achieve comparable fidelity to expensive multi-view rigs using only *two* physical camera views.
- Concluding, Chap. 6 opens the discussion concerning the ramifications of the proposed work as well as some immediate potential future directions that could be investigated on top of the proposed work.

Chapter 2

Introduction

Triangulation refers to determining the location of a point in 3D space from projected 2D correspondences across multiple views. In theory, only *two* calibrated camera views should be necessary to accurately reconstruct the 3D position of a point. However, in practice the effectiveness of triangulation is heavily dependent upon the accuracy of the measured 2D correspondences, baseline, and occlusions. As a result expensive and cumbersome multi-view rigs, sometimes involving hundreds of cameras and specialized hardware, are currently the method of choice to obtain high fidelity 3D reconstructions of non-rigid objects [19].

In this thesis, we challenge the need to have such complicated multi-view rigs to obtain high-fidelity 3D reconstructions. We show that comparable fidelity to these high complexity rigs can be obtained using as little as *two* uncalibrated views. Such a simplification would enable data collection in unstructured, "in-the-wild" environments, opening the door to a wide variety of applications ranging from entertainment, neuroscience, psychology, ethology, and several fields of medicine [7, 10, 12, 15, 22], where complex multi-camera rigs may be financially, technologically, or simply practically infeasible.

One of the most notable multi-view rigs for human pose reconstruction is the PanOptic studio [19], which contained 480 VGA cameras, 31 HD Cameras, and 10 RGB+D sensors, distributed over the surface of geodesic sphere with a 5.49m diameter. This setup also required specialized hardware for storage and gen-lock camera exposures. Despite its cost and complexity, the fidelity of the 3D reconstructions from PanOptic

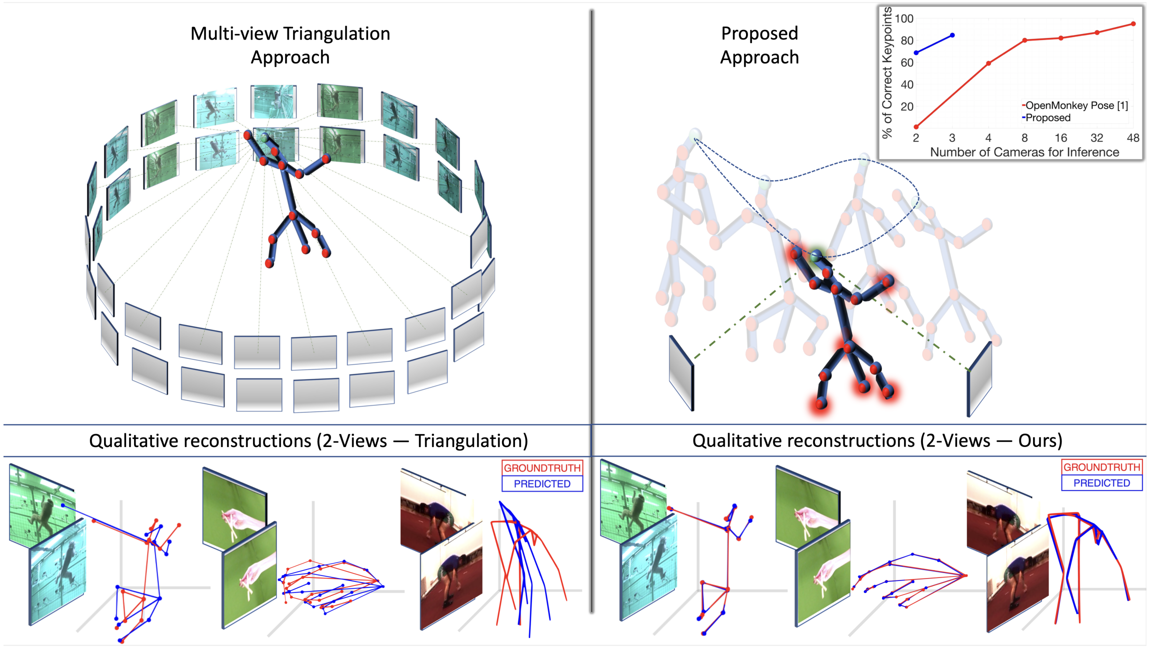

Figure 2.1: A traditional multi-view setup relies on the concept of triangulation with the assumption that the point being reconstructed is static in time – requiring a large number of physical views (i.e. cameras) to ensure a high fidelity reconstruction. Our approach breaks this triangulation assumption by allowing the reconstructed points to deform according to a neural shape prior. Empirically (see the plot in top-right), we demonstrate that our approach can achieve comparable fidelity to expensive multi-view rigs using only two physical views. Blue lines depict the triangulation and proposed approaches (left vs. right, respectively) with as little as two-physical views and red lines show the corresponding 3D ground-truth.

studio has motivated similar efforts across industry and academia. Of particular note is a recent effort that employed 62 hardware synchronized cameras to capture the pose of Rhesus Macaque monkeys [4]. Other notable efforts include [21] for dogs, [16] for human body, and [11, 41] for the human face.

The idea of applying Non-Rigid Structure from Motion (NRSfM) to reduce the number of physical cameras required to reconstruct an object is not new. Numerous works exist on the application of NRSfM to this problem – predominantly using a single physical camera. They all rely on replacing classical rigidity with other shape constraints notably: *(i)* low rank [2, 6, 8, 9, 30], *(ii)* union-of-subspaces [1, 31, 50], and *(iii)* compressibility [25, 27, 49]. These constraints are especially problematic when it comes to high-fidelity reconstructions as they can only be applied to certain types of objects with limited amounts of non-rigidity, or they are hyper-sensitive to noise in the 2D correspondences. Temporal constraints [3, 29] can help with this, but they can only be applied to temporal sequences with well understood dynamics. Recently, neural shape priors have been developed in single-view NRSfM [26], and have been demonstrated to significantly outperform these classical priors in terms of modeling shape complexity and sensitivity to noise.

In this work, we dramatically reduce the required number of cameras while removing calibration requirements for multi-view rigs such as [19] by further constraining the neural shape prior approach with a view-equivariance assumption – namely that multiple simultaneous views correspond to a single shape – thus preserving the benefits of both multi-view and neural-prior constraints within our proposed NRSfM framework. Figure 2.1 presents a graphical depiction of our approach. To our knowledge, this paper is the first effort to apply neural priors to multi-view NRSfM.

Contributions: In this thesis, we make two major contributions. First, we propose a multi-view NRSfM architecture that incorporates a neural shape prior while enforcing equivariant view consistency. Second, we demonstrate that this framework is competitive with some of the most complicated multi-view capture rigs – while only requiring a modest number (2-3) of physical camera views. Our effort is the first we are aware of that utilizes these new advances in neural shape priors for multi-view 3D reconstruction. Figure 2.1 presents a graphical depiction of our approach. Extensive evaluations are presented across numerous benchmarks and object categories including the human body, human hands, and monkey body. We should note that our proposed approach assumes known 2D projected measurements so does not directly leverage pixel intensities. Our approach, however, can be integrated with any available 2D landmark image detector such as HR-Net [43], Stacked Hourglass Networks [34], Integral Pose Regression [44], and others.

Chapter 3

Related Work

3.1 Multi-view approaches:

Multi-view triangulation [13] has been the method of choice in the context of large-scale complex rigs with multiple cameras [4, 11, 19, 41] for obtaining 3D reconstruction from 2D measurements. The number of views, 2D measurement noise, baseline, and occlusions bound the fidelity of these 3D reconstructions. These time-synchronized multiple physical views also come at considerable cost and effort.

Recent work by Iskakov et al. [18] and others [20, 38, 39, 46] have explored how supervised learning can be used to enhance multi-view reconstruction. Similarly, work by Rhodin et al. [40] and Kacobas et al. [24] attempted to use supervised and self-supervised learning, respectively, to infer 3D geometry from a single physical camera view. An obvious drawback to these approaches is that one is required to have intimate 3D supervision of the object before deployment – a limitation that modern multi-view rigs are not faced with. None of these approaches are as general as the one we are proposing. For example, nearly all these prior works deal solely with the reconstruction of the human pose as they are heavily reliant upon peripheral 3D supervision.

3D supervision	Iskakov et al. [18]
	Remelli et al. [39]
	Kadkhodamohammadi et al. [20]
	Tome et al. [46]
	Pavlakos et al. [38]
	Multi-view Martinez [33]
	Rhodin et al. [40]
	Kocabas et al [24]
Unsupervised	Kocabas et al. (SS w/o R) [24]
	PRN [37]
	RepNet [47]
	Iqbal et al. [17]
	Pose-GAN [28]
	Deep NRSfM [26]
	C3DPO [35]
	MV NRSfM (Ours)

Figure 3.1: Red tint rows have 3D supervision. Green tint are unsupervised 3D reconstruction methods.

3.2 Monocular NRSfM:

Of particular interest in this paper is to utilize NRSfM approaches that are atemporal. Modern multi-view rigs make no assumptions about the dynamics of the object they are reconstructing – we want our approach to have similarly broad applicability. The NRSfM task [6] is to simultaneously recover the non-rigid 3D structure and camera pose from an ensemble of 2D measurements captured at different points in time. Advances in unsupervised learning based approaches to NRSfM [26, 35] have seen significant improvements in their robustness and fidelity across a broad set of object categories and scenarios. These recent advances to date have only been applied to problems where there is only a single view (i.e. monocular) of the object at a particular point in time. Our approach is the first – to our knowledge – to leverage these advancements for 3D reconstruction when there are multi-view measurements taken at the same instance in time.

3.3 Amount of supervision:

There exists now several multi-view approaches for 3D human pose estimation that leverage either full or weak 3D supervision [18, 20, 24, 33, 38, 39, 40, 46]. None of these references, however, directly tackle the unsupervised multi-view 3D reconstruction problem and hence are not as general as our solution. These approaches however, could be utilized as part of literature review for showcasing the generalization capability of the proposed approach. The supervised approaches are shown with a green tint in Fig. 3.1. Furthermore, the unsupervised approaches such as [17, 26, 28, 35, 37, 47] are shown with red tint in Fig. 3.1. This table shows the recent monocular unsupervised 3D reconstruction methods as well in the lower red tint part as well.

Chapter 4

Multi-view Neural Shape Prior

Problem setup. We are interested in a camera rig setup with K synchronized views capturing N instances of non-rigid objects from the same category. Specifically, we are given a non-sequential (atemporal) dataset containing N multi-view 2D observations $\{\mathbf{W}_1^{(1)}, \ldots, \mathbf{W}_1^{(N)}; \cdots ; \mathbf{W}_K^{(1)}, \ldots, \mathbf{W}_K^{(N)}\}$, where each $\mathbf{W} \in \mathbb{R}^{P\times 2}$ represents 2D location for P keypoints. We want to reconstruct the 3D shape $\mathbf{S}^{(1)}, \ldots, \mathbf{S}^{(N)}$, where each $\mathbf{S} \in \mathbb{R}^{P\times 3}$ for each of the N instances of the object.

Weak perspective projection. We assume weak perspective projections, *i.e.* for a 3D structure $\mathbf{S}$ defined at a canonical frame, its 2D projection is approximated as

$$\mathbf{W} \approx s\mathbf{S}\mathbf{R}_{xy} + \mathbf{t}_{xy} \tag{4.1}$$

where $\mathbf{R}_{xy} \in \mathbb{R}^{3\times 2}$, $\mathbf{t}_{xy} \in \mathbb{R}^2$ are the x-y component of a rigid transformation, and $s > 0$ is the scaling factor inversely proportional to the object depth if the true camera model is pin-hole. If all 2D points are visible and centered, $\mathbf{t}_{xy}$ can be omitted by assuming the origin of the canonical frame is at the center of the object. Due to the bilinear form of (4.1), s is ambiguous and becomes up-to-scale recoverable only when $\mathbf{S}$ is assumed to follow certain prior statistics. In our approach, we handle scale by approximating with an orthogonal projection and solving an Orthogonal-N-Point (OnP) problem [42] to find the camera pose along with the scale, as discussed in Sec. 4.1.2.

Statistical shape model. We assume a linear model for the 3D shapes $\mathbf{S}$ to be reconstructed, *i.e.* at canonical coordinates, the vectorization of $\mathbf{S}$ in Eq. (4.1),

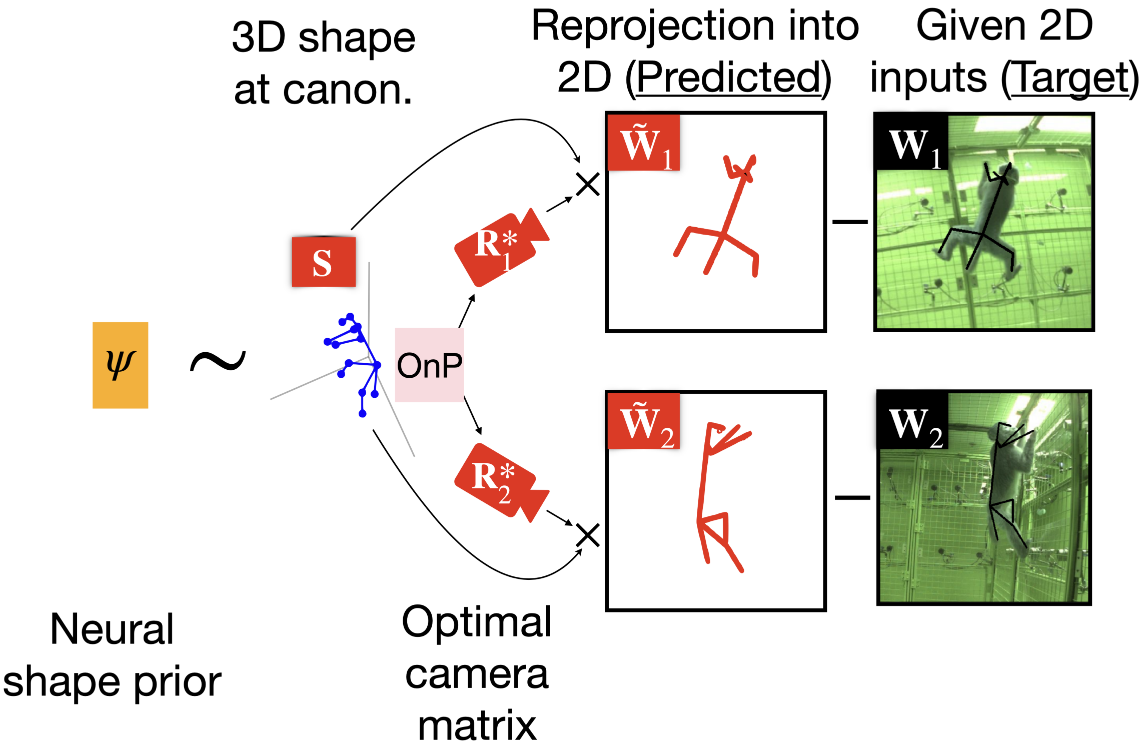

Figure 4.1: Two views statistical shape prior. The 3D structure $\mathbf{S}$ is drawn from a statistical shape distribution using neural shape priors and consequently projected to 2 views using the cameras $\mathbf{R}_k^*$ $\forall k \in [1, 2]$ – calculated through OnP formulation [42]. The proposed approach minimizes the 2D projection error between the predicted 2D projections $\tilde{\mathbf{W}}_k$ and target (input) 2D projections $\mathbf{W}_k$.

denoted $\mathbf{s} = \text{vec}(\mathbf{S}) \in \mathbb{R}^{3P}$ can be written as

$$\mathbf{s} = \mathbf{D}\boldsymbol{\psi} \tag{4.2}$$

where $\mathbf{D} \in \mathbb{R}^{3P \times B}$ is the shape dictionary with B basis and $\boldsymbol{\psi} \in \mathbb{R}^B$ is the code vector - taking insight from classical sparse dictionary learning methods. The factorization of $\mathbf{S}$ in Eq. (4.2) is ill-posed by nature; in order to resolve the ambiguities in this factorization, additional priors are necessary to guarantee the uniqueness of the solution. Notable priors include the assumption of $\mathbf{S}$ being (i) low rank [2, 6, 8, 9, 30], (ii)lying in a union-of-subspaces [1, 31, 50] (iii) or compressible [25, 27, 49]. The low-rank assumption becomes infeasible when the data exhibits complex shape variations, the union-of-subspaces NRSfM methods have difficulty clustering shape deformations and estimating affinity matrices effectively. Finally, the sparsity prior allows more powerful modeling of shape variations with large number of subspaces, but suffers from sensitivity to noise.

Neural Shape Prior Our neural shape prior is an approximation to a hierarchical sparsity prior introduced by Kong et al. [26], where each non-rigid shape is represented by a sequence of hierarchical dictionaries and corresponding sparse codes. Other neural shape priors – such as C3PDO [35] – could be entertained as well but we chose to employ Kong et al.'s method due to its simplicity with respect to enforcing multi-view equivariant constraints. The approach in [26] maintains the robustness of sparse code recovery by utilizing overcomplete dictionaries to model highly deformable objects consisting of large-scale shape variation. Moreover, if the subsequent dictionaries in this multi-layered representation are learned properly, they can serve as a filter such that only functional subspaces remain and the redundant are removed. Due to the introduction of multiple levels of dictionaries and codes in the following section, we will abuse the notation of $\mathbf{D}, \boldsymbol{\psi}$ by adding left superscript 1, *i.e.* ${}^{1}\mathbf{D}$, ${}^{1}\boldsymbol{\psi}$ indicating that they form the first level of hierarchy. Assuming the canonical 3D shapes are compressible via multi-layered sparse coding with $l \in L$ layers, the shape code ${}^{1}\boldsymbol{\psi}$ is constrained as

$$\begin{aligned} \mathbf{s} &= {}^{1}\mathbf{D}\,{}^{1}\boldsymbol{\psi} \\ {}^{1}\boldsymbol{\psi} &= {}^{2}\mathbf{D}^{2}\boldsymbol{\psi} \\ &\vdots \\ {}^{L-1}\boldsymbol{\psi} &= {}^{L}\mathbf{D}^{L}\boldsymbol{\psi} \\ \text{s.t.} \quad \|{}^{l}\boldsymbol{\psi}\|_1 &\le {}^{l}\boldsymbol{\lambda}\ ,\ {}^{l}\boldsymbol{\psi} \ge \mathbf{0}\ ,\ \forall l \in \{1, \ldots, L\} \end{aligned} \tag{4.3}$$

where ${}^{l}\mathbf{D} \in \mathbb{R}^{{}^{l-1}B \times {}^{l}B}$ are the hierarchical dictionaries, l is the index of hierarchy level, and ${}^{l}\boldsymbol{\lambda}$ is the scalar specifying the amount of sparsity in each level. Thus, the learnable parameters are $\Theta = \{\cdots, {}^{l}\mathbf{D}, {}^{l}\boldsymbol{\lambda}, \cdots\}$. The single set of parameters Θ are fit *jointly* along with the sparse codes, rotation matrices, and structures $\mathbf{S}$ for each instance in the dataset. Jointly constraining each instance via a common set of weights (the "neural prior") makes this work more akin to classic factorization methods, in which both the shared factors and the weightings for each instance are jointly inferred, rather than to network training approaches which aim to find weights that generalize well when later used to perform inference on unseen data.

Factorization-based NRSfM. Equivalently, the linear model in Eq. (4.2) could be rewritten as

$$\mathbf{S} = \mathbf{D}^{\#}(\boldsymbol{\psi} \otimes \mathbf{I}_3)$$

where $\mathbf{D}^{\#} \in \mathbb{R}^{P \times 3B}$ is a reshape of $\mathbf{D}$ and $\otimes$ denotes a Kronecker product. Applying the camera matrix $\mathbf{R}_{xy}$ gives the 2D pose. Thus

$$\mathbf{S}\mathbf{R}_{xy} = \mathbf{D}^{\#}(\boldsymbol{\psi} \otimes \mathbf{R}_{xy})$$

Substituting the input 2D pose $\mathbf{W}$ from Eq. (4.1), we have

$$\begin{aligned} &\mathbf{W} = \mathbf{D}^{\#}\boldsymbol{\Psi}_{xy} \\ \text{s.t.} \quad &\boldsymbol{\Psi}_{xy} = \boldsymbol{\psi} \otimes \mathbf{R}_{xy} \text{ and } \boldsymbol{\psi} \in \mathcal{C} \end{aligned} \tag{4.4}$$

where $\boldsymbol{\Psi}_{xy} \in \mathbb{R}^{3B \times 2}$ is the sparse block code denoting the first two columns of $\boldsymbol{\Psi} \in \mathbb{R}^{3B \times 3}$; and $\mathcal{C}$ denotes the neural shape prior constraints applied on the code $\boldsymbol{\psi}$, *e.g.* hierarchical sparsity [26] in our case. Conceptually, $\boldsymbol{\Psi}$ is a matrix with rotations and sparse code built into it. Under the unsupervised settings, $\mathbf{D}, \boldsymbol{\psi}, \mathbf{R}, \mathbf{S}$ are all unknowns and are solved under the simplified assumptions that the input 2D poses are obtained through a weak perspective or an orthogonal camera projection. It is important to note that along with the $\mathbf{R}$ predicted through the factorization of $\boldsymbol{\Psi}$, we also analytically compute $\mathbf{R}^*$ as a solution to a Orthographic-n-point (OnP) problem that implicitly acts as a supervisory signal for the $\mathbf{R}$ generated from $\boldsymbol{\Psi}$. Corresponding proof for $\mathbf{R}^*$ is discussed in the supplementary section.

4.1 Approach

4.1.1 Bilevel optimization

Given only the input 2D poses in Eq. (4.4), two problems remain to address

- How to formulate an optimization strategy to recover $\mathbf{D}, \boldsymbol{\psi}, \mathbf{R}, \mathbf{S}$?

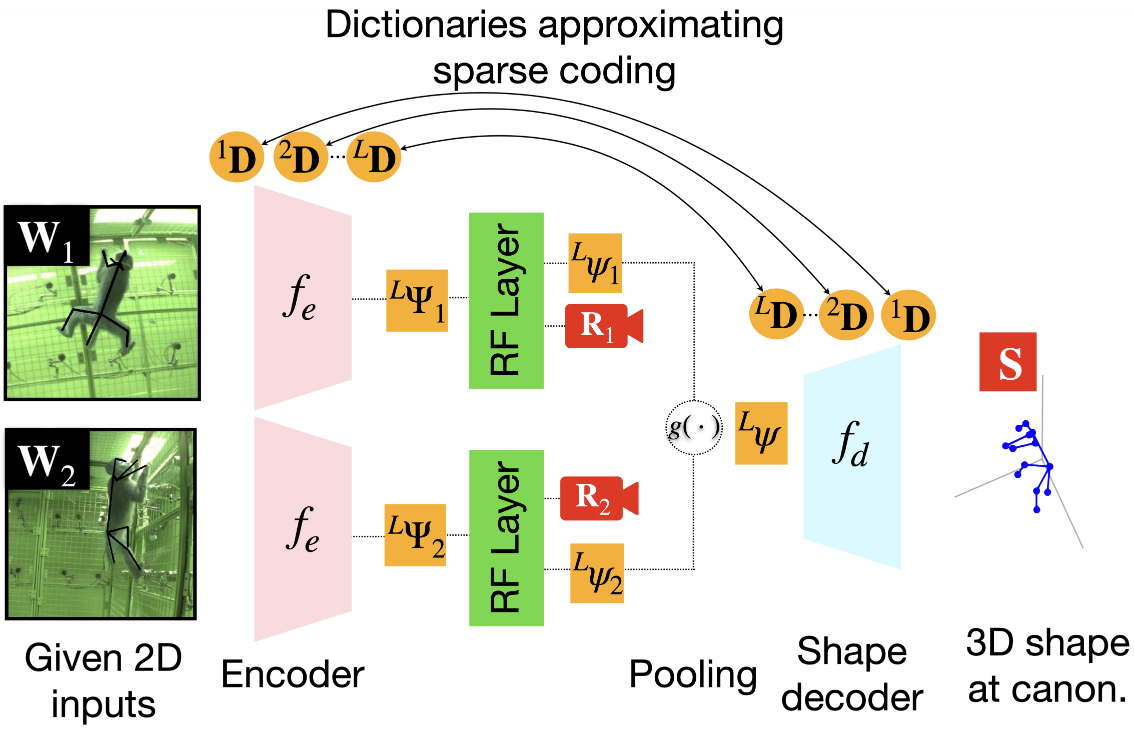

Figure 4.2: Architecture showing our $K = 2$-views 3D reconstruction approach. The 2D projections from both views $\mathbf{W}_k$ $\forall k \in [1, 2]$ acts as an input to encoder $\boldsymbol{f}_e$ that extracts the block sparse code $\boldsymbol{\Psi}_k$ from the corresponding views. A Rotation Factorization (RF) layer at the bottleneck stage shown in green, factorizes the block sparse code into the respective camera matrix $\mathbf{R}_k$ and the unrotated vector sparse code ${}^L\boldsymbol{\psi}_k$. The codes are then fused via *pooling* function $\boldsymbol{g}$ into a single code ${}^L\boldsymbol{\psi}$ that acts as an input to the shape decoder $\boldsymbol{f}_d$. The shape decoder predicts the 3D structure $\mathbf{S}$ in the canonical frame while enforcing equivariant view consistency.

- How to efficiently pool in K different camera views and enforce equivariance over the predicted K camera matrices and a single 3D structure in canonical frame?

We choose to impose neural shape priors through hierarchical sparsity constraints [26] literature for approaching a solution to the above problems, with learnable parameters Θ (see Sec. 4.1.2). From Eq. (4.4), the learning strategy of multi-view NRSfM problem for N instances with K views is then interpreted as solving the following bilevel optimization problem. Eq. (4.3) leads to relaxation of the following lower-level problem

$$\begin{aligned}\min_{\mathbf{D},\Theta}\sum_{k=1}^{K}\sum_{n=1}^{N}\Bigg(&\min_{{}^{l}\boldsymbol{\psi}_k^{(n)},\mathbf{R}_k^{(n)}}\|\mathbf{W}_k^{(n)}-{}^{1}\mathbf{D}\big({}^{1}\boldsymbol{\Psi}_k^{(n)}\big)\|_F+\\ &\sum_{l=1}^{L}{}^{l}\boldsymbol{\lambda}\|{}^{l}\boldsymbol{\Psi}_k^{(n)}\|_F+\sum_{l=2}^{L}\|({}^{l-1}\boldsymbol{\Psi}_k^{(n)})-{}^{l}\mathbf{D}\,({}^{l}\boldsymbol{\Psi}_k^{(n)})\|_F\Bigg)\end{aligned} \tag{4.5}$$

where the first expression in (4.5) minimizes the 2D projection error, the second expression enforces sparsity, and the third expression fits each dictionary in the hierarchy to the dictionary representation in the preceding layer.

4.1.2 Network approximate solution

The optimization problem in Eq. (4.5) is an instance of dictionary learning problem with sparse codes $\boldsymbol{\psi}$. The classical approach to this problem is by solving the Iterative Shrinkage and Thresholding Algorithm (ISTA) [5]. However, Papyan et al. [36] show that a single layer feedforward network with Rectified Linear Unit (ReLU) activations approximate one step of ISTA, with the bias terms ${}^{l}\boldsymbol{\lambda}$ adjusting the sparsity of recovered code for the l^{th} layer. Furthermore, the dictionaries $[{}^{1}\mathbf{D},\ldots,{}^{L}\mathbf{D}]$ can be learned by back-propagating through the feedforward network. We devise a network architecture that serves as an approximate solver to the above optimization problem and provide derivations in the following subsections.

Approximating sparse codes. We review the sparse dictionary learning problem and consider the single-layer case stated above. To reconstruct an input signal $\mathbf{X}$, the optimization problem becomes

$$\min_{\boldsymbol{\Psi}}\|\mathbf{X}-\mathbf{D}\boldsymbol{\Psi}\|_F+\boldsymbol{\lambda}\|\boldsymbol{\Psi}\|_F$$

As stated above, Papyan et al. [36] propose that one iteration of ISTA gives back the block-sparse codes $\boldsymbol{\Psi}$ as

$$\boldsymbol{\Psi}=\mathrm{ReLU}(\mathbf{D}^{\top}\mathbf{X};\boldsymbol{\lambda})$$

We interpret ReLU as solving for the block-sparse code and incorporate ReLU as the nonlinearity in our encoder part of the network.

Encoder architecture. We propose to devise an encoder network f_e that takes the 2D poses as input and outputs the block sparse codes $\boldsymbol{\Psi}$ that has within itself the rotation matrix $\mathbf{R}$ as well as a rotationally invariant sparse code $\boldsymbol{\psi}$, *i.e.* $f_e(\mathbf{W}_k^{(n)}) \mapsto \Big({}^L\boldsymbol{\Psi}_k^*\Big)$. Unrolling one iteration of ISTA for each layer, f_e takes $\mathbf{W}_k^{(n)}$ as 2D pose inputs and produces block sparse codes for the last layer $[\,{}^1\boldsymbol{\Psi}_k^{(n)}, \ldots, {}^L\boldsymbol{\Psi}_k^{(n)}]$ as output, shown in Fig. A.1

$$\begin{aligned}
{}^1\boldsymbol{\Psi}_k^{(n)} &= \text{ReLU}\Big(\big[({}^1\mathbf{D}^{\#})^\top \cdot \mathbf{W}_k^{(n)}\big]_{3\times 2};\, {}^1\boldsymbol{\lambda}^{(n)}\Big) \\
{}^2\boldsymbol{\Psi}_k^{(n)} &= \text{ReLU}\Big(({}^2\mathbf{D} \otimes \mathbf{I}_3)^\top \cdot {}^1\boldsymbol{\Psi}_k^{(n)};\, {}^2\boldsymbol{\lambda}^{(n)}\Big) \\
&\ \vdots \\
{}^L\boldsymbol{\Psi}_k^{(n)} &= \text{ReLU}\Big(({}^L\mathbf{D} \otimes \mathbf{I}_3)^\top \cdot {}^{L-1}\boldsymbol{\Psi}_k^{(n)};\, {}^L\boldsymbol{\lambda}^{(n)}\Big)
\end{aligned} \tag{4.6}$$

where ${}^l\boldsymbol{\lambda}^{(n)}$ is the learnable threshold for each layer. $({}^l\mathbf{D}\otimes\mathbf{I}_3)^\top \cdot {}^{l-1}\boldsymbol{\psi}_k^{(n)}$ is implemented by a convolution transpose.

Rotation Factorization layer. At the bottleneck, our encoder network generates a block sparse code for $K-$views ${}^L\boldsymbol{\Psi}_k^{(n)}$. As evident in Eq. (4.4), since the block sparse code has rotations $\mathbf{R}_k^{(n)}$ as well as an unrotated sparse code $\boldsymbol{\psi}_k^{(n)}$, we add a fully-connected layer that factorizes out these quantities, named Rotation Factorization (RF) layer, shown as a green block in Fig. A.1. Consequently, ${}^L\boldsymbol{\Psi}_k^{(n)}$ is then factorized into an unrotated sparse code ${}^L\boldsymbol{\psi}_k^{(n)}$ and the rotation matrix $\mathbf{R}_k^{(n)}$ (constraining to $SO(3)$ using SVD) using this fully-connected RF layer. At this stage, we pool the features from all the rotationally invariant or unrotated sparse codes ${}^L\boldsymbol{\psi}_k^{(n)}$ using a sum pooling operation $\boldsymbol{g}$ that enforces the equivariance consistency within all the views by combining features from multiple views.

$$\boldsymbol{g}({}^L\boldsymbol{\psi}_1^{(n)}, \ldots, {}^L\boldsymbol{\psi}_K^{(n)}) \mapsto ({}^L\boldsymbol{\psi}^{(n)}) \tag{4.7}$$

as shown in architecture overview Fig. A.1, where $\boldsymbol{g}$ denotes a **sum** operation. Since the pooled sparse code ${}^L\boldsymbol{\psi}$ is rotationally invariant, we generate a single canonical 3D structure $\mathbf{S}$ through a decoder network $\boldsymbol{f}_d$, that remains equivariant to

K camera rotations $\mathbf{R}_1, \ldots, \mathbf{R}_K$. Thus, the decoder network $\boldsymbol{f}_d$ helps supervise the fully-connected RF layer.

Insight behind multi-view consistency. For each individual view, we get a block sparse code representation $\boldsymbol{\Psi}_k^{(n)}$ that has the rotation $\mathbf{R}_k^{(n)}$ combined with an unrotated sparse code $\boldsymbol{\psi}_k^{(n)}$. RF layer disentangles these quantities and generates codes that are consistent with an unrotated or canonicalized view. This architecture thus enforces equivariance consistency by consequently passing the unrotated sparse code $\boldsymbol{\psi}$ through a shape decoder to produce a canonicalized 3D structure. When we jointly encode multiple views into a single canonical shape, the equivariance is implicitly enforced after projecting them through the given multiple cameras. These multi-view projections help supervise the multi-view NRSfM network.

Decoder architecture. Finally, a decoder $\boldsymbol{f}_d$ is devised that takes input a pooled bottleneck sparse code (see Eq. 4.7) and generates a canonical 3D structure $\mathbf{S}$. Thus, $f_d({}^{L}\boldsymbol{\psi}^{(n)}) \mapsto \left(\mathbf{S}^{(n)}\right)$

$$\begin{aligned}
{}^{L-1}\boldsymbol{\psi}^{(n)} &= \mathrm{ReLU}({}^{L}\mathbf{D} \cdot {}^{L}\boldsymbol{\psi}^{(n)}; {}^{L}\boldsymbol{\lambda}^{(n)}) \\
&\vdots \\
{}^{1}\boldsymbol{\psi}^{(n)} &= \mathrm{ReLU}({}^{2}\mathbf{D} \cdot {}^{2}\boldsymbol{\psi}^{(n)}; {}^{2}\boldsymbol{\lambda}^{(n)}) \\
\mathbf{S}^{(n)} &= {}^{1}\mathbf{D} \cdot {}^{1}\boldsymbol{\psi}^{(n)}
\end{aligned} \tag{4.8}$$

We analytically compute a closed-form solution to $\mathbf{R}^*$ as a solution to an Orthographic-n-point (OnP) problem that implicitly acts as supervisory signal for the $\mathbf{R}_k^{(n)}$. Detailed proof is shown in the supplementary section.

Supervising R using solution from OnP We are using a closed-form solution to $\mathbf{R}^*$ that gives us an optimal solution different from the one produced by the network at the bottleneck stage. We opt to use an algebraic solution which can be implemented as a differentiable operator and could be easily accomplished via modern autograde packages. The $\mathbf{R}^*$ generated by OnP implicitly supervises the $\mathbf{R}$ generated in the RF layer of the encoder-decoder network. The detailed proof for the OnP solution is given in the appendix.

Loss function To reemphasize the loss function in our neural architecture, the loss function driving the proposed approach is a reprojection error

$$\mathcal{L} = \frac{1}{KN} \sum_{k=1}^{K} \sum_{n=1}^{N} \| \mathbf{W}_k^{(n)} - \mathbf{S}^{(n)} \mathbf{R}_k^{(n)} \|_F \tag{4.9}$$

Chapter 5

Results – Multi-view Neural Shape Prior

5.1 Experiments

We present evaluations across numerous object categories including the human body, human hands, and monkey body. Evaluation is divided into two major categories: *(i)* Multi-view 3D reconstruction of an input 2D dataset, and *(ii)* Generation of 3D labels for unseen 2D data. The former compares against classical algorithms to generate high-fidelity 3D reconstruction from multi-view 2D input datasets. The latter discusses the generalization capability of our approach, and shows that it does not overfit. For this we follow one of the standard protocols for a human pose dataset and show results on the validation split. We go through the experimentation and dataset details.

Network architecture and implementation details In our implementation, we use the same neural encoder architecture for all the K views across different datasets. As shown in Fig. A.1 we use $K-$encoders and a single shape decoder to generate one 3D structure in a canonicalized frame. The dictionary size (*i.e.* neural units) within each layer of encoder is decreased exponentially: $\{1024, 512, 256, 128, 64, 32, 16, 8\}$. Ideally, if a validation set with 3D groundtruth is provided, we could select optimal architecture based on cross-validation. However, due to the unsupervised setting, we rather set the hyperparameters heuristically. We pick a bottleneck dimension, 8 for articulated objects such as human skeleton, human hands, or monkey body. For the encoder and decoder architecture discussed in Eq. (4.6), (4.8), we use a convolutional network as in Kong et al. [26] and share the convolution kernels (*i.e.* dictionaries) between the encoder and decoder.

Training details We keep the same weightings for the reprojection error shown in loss function Eq. (4.9). We use the Adam optimizer [23] in our implementation.

Evaluation metrics We utilize the following metrics to assess the prediction accuracy of 3D reconstruction. **PA-MPJPE**: prior to computing the mean per-joint position error, we standardize the scale of the predictions by normalizing them to match against the given ground-truth (GT) followed by rigidly aligning these predictions to GT. Lower the better **PCK**: percentage of correct keypoints. The predicted joint is viewed as correct if the separation between the predicted and the GT joint is within a specific range (usually in *cm* or *mm*).

Monkey body dataset OpenMonkeyStudio [4] is a huge Rhesus Macaque monkey pose dataset in a setup similar to PanOptic Studio where 62 cameras capture the markerless pose of Rhesus Macaque monkeys. We use the provided 2D annotations over the Batch (7, 9, 9a, 9b, 10, and 11). This dataset also provides the groundtruth 3D labels for the given batches to evaluate the 3D reconstruction performance.

Human body dataset Human 3.6 Million (H3.6M) [16] is a large-scale human pose dataset with 3.6 million images featuring 11 actors performing 15 daily activities, such as eating, sitting, walking, and taking a photo, from 4 camera views - annotated by motion capture systems. The 2D keypoint annotations of H3.6M preserve the perspective effect, and thus is a realistic dataset for evaluating the practical usage of generating 3D labels for unseen data as well as test the generalization capability of our approach. We use this dataset for both quantitative and qualitative evaluation. For generating 3D reconstruction of an input dataset (see Sec. 5.1.1), we pick 5 subjects (1, 5, 6, 7, 8) and compare against the classical multi-view triangulation baselines. For generating 3D labels over unseen 2D data to showcase the generalization capability (see Sec. 5.1.2), we follow the standard protocol on H3.6M and use the subjects (1, 5, 6, 7, 8) during the training stage and the subjects (9, 11) for evaluation stage.

Evaluation is performed on every 64th frame of the test set. We include average errors for each method.

Human hands dataset Finally, we use an open-source hands dataset - FreiHand [51] - a large-scale open-source dataset with varied movements of hands with 3D pose annotated by motion capture systems. It consists of 32560 samples with their corresponding camera intrinsics. We generate random camera extrinsics and randomly create multiple camera views to generate multi-view 2D inputs for evaluating the proposed approach.

5.1.1 3D reconstruction of an input 2D dataset

Like classical 3D reconstruction algorithms such as multi-view triangulation or bundle adjustment, our approach jointly infers the unknown 3D shape and camera rotations from 2D keypoints. By simultaneously fitting the shared network parameters used to recover shape and pose, our approach constrains the possible reconstructions much more strongly than multi-view triangulation or bundle-adjustment approaches. We emphasize that, while we later showcase the generalization capability of the setup by applying the fitted network to generate 3D labels for unseen 2D data (see Sec. 5.1.2), the major contribution of our approach is the optimization process for multi-view 3D reconstruction of an input 2D dataset. The goal is to evaluate the robustness of the proposed multi-view neural shape prior across different shape variations and hence as part of the evaluation, we report how well our method is able to reconstruct different datasets compared to the baseline methods.

Baseline We use a baseline implementation of iterative multi-view triangulation with robust outlier rejection [13, 14], referred to as **TRNG**. This is also the triangulation method of choice for recent multi-view 3D human pose learning by Kocabas et al. [24], who also provide an open-source implementation for this method. A more recent method doing classical optimization on triangulation is proposed by Lee and Civera [32], however, their method is not necessarily optimal in terms of accuracy, but more in terms of computation time. **TRNG** first finds the points which minimize the distance from all the rays and removes the rays which are the furthest away from that point. It then re-evaluates the triangulation and this iteration is repeated 2-3 times. Empirically, we find that increasing the iteration leads us to predict near-perfect

Method	Batch#7	Batch#9	Batch#9a	Batch#9b	Batch#10	Batch#11
TRNG	21.21	24.32	30.67	24.50	26.10	22.77
MV NRSfM	**8.36**	**8.25**	**9.12**	**11.52**	**8.203**	**8.17**

Table 5.1: **PA-MPJPE** error values for Monkey body dataset shows substantial improvement over the baseline rigid multi-view triangulation approach while using only two views over noisy 2D keypoints. **PA-MPJPE** values are in **cm**.

	S1, S5, S6, S7, S8								
	Extrinsics Noise			Intrinsics Noise			2D keypoints Noise		
	$\sigma = 0.1$	$\sigma = 0.5$	$\sigma = 0.9$	$\sigma = 0.1$	$\sigma = 0.5$	$\sigma = 0.9$	$\sigma = 15$	$\sigma = 25$	$\sigma = 35$
TRNG	65.49	131.66	145.94	69.57	188.63	234.47	70.08	114.06	154.41
2-Views (ours)	**30.53**						**54.22**	**65.74**	**77.82**

Table 5.2: Robustness to camera calibration and 2D annotations noise for Human 3.6M dataset.

3D reconstruction if we have exact camera calibration parameters and exact, clean 2D projections. We consider this to be a very strong baseline comparison since this approach is being widely used in industry as well as academia to generate very accurate 3D reconstructions that are further used to train 3D regression methods. The detailed proof is provided in the supplementary section. We evaluate our approach on the above three datasets with substantial non-rigid deformities. For all the given experiments the 2-view cameras are chosen at random and the same set of cameras are used in the comparative baselines for a fair comparison.

Method	**PCK**
2 Views [4]	1.2%
4 Views [4]	59%
8 Views [4]	80%
16 Views [4]	82%
32 Views[4]	87%
48 Views [4]	95%
2-views (ours)	**68.63**%
3-views (ours)	**84.63**%

Table 5.3: Percentage of Correct Keypoint (PCK) % for OpenMonkeyStudio dataset. Following [4], the threshold for considering a keypoint to be correct is set at $10cm$.

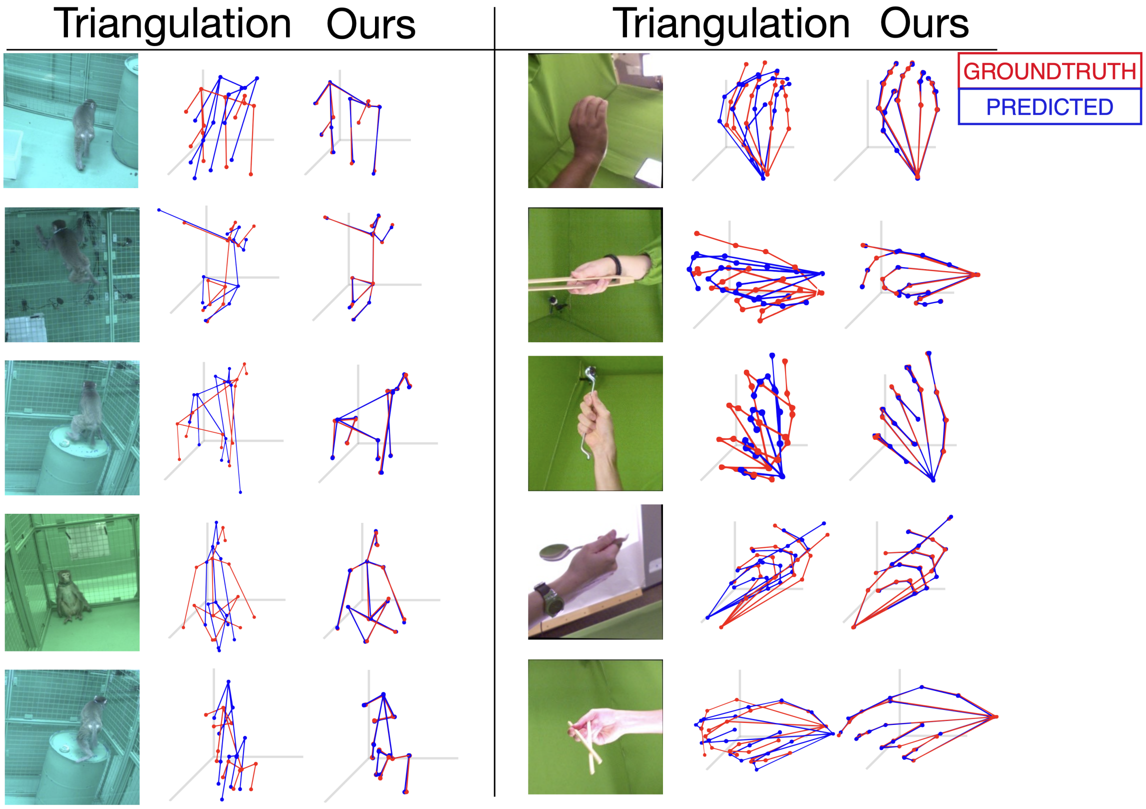

Figure 5.1: Qualitative 3D reconstruction comparison between the multi-view triangulation technique and our technique for Monkey body [4] and human hands [51] when operated over noisy 2D keypoints.

Evaluation analysis For the Monkey body dataset, multi-view 3D reconstruction with 2− or 3− view using our approach significantly outperforms the given results in [4] and achieves comparable fidelity with only two physical views. Similar to [4], we consider all the keypoints as correct if their reconstruction is within $10cm$ of the groundtruth in the **PCK** protocol. Table 5.3 and top-right plot in Fig. 2.1 shows that we outperform the given results of 2-Views by a significant margin (1.2% vs. 68.63%). The fidelity of 3D reconstructions using the proposed method continues to rise as we add in more views - evident by the uptick in performance from 3−views. Qualitative performance of Monkey dataset and human hands dataset is shown in Fig. 5.1.1 and quantitative performance of monkey body is given in Tab. 5.1 when operated over noisy 2D keypoints. For the human body dataset, we inject noise in the camera extrinsics, intrinsics, and 2D keypoints separately and compare the

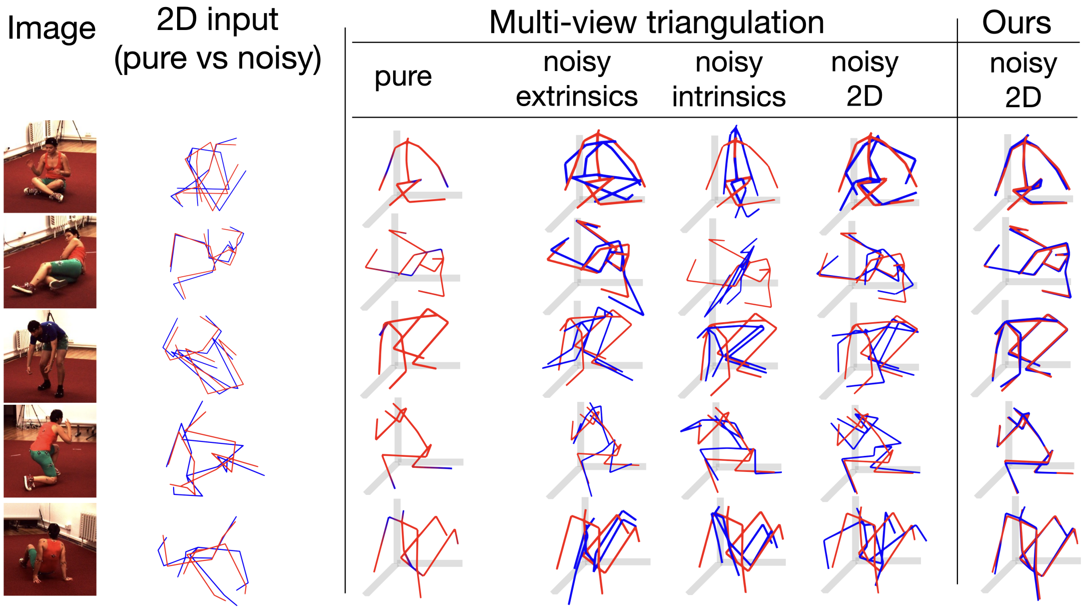

Figure 5.2: Qualitative results on Human 3.6M dataset with $\sigma = [0.5, 0.5, 25]$ as intrinsics, extrinsics, and 2D keypoints Gaussian noise, respectively.

performance in Fig. 5.2 and Tab. 5.2. The baseline method fails when noise with a small standard deviation is added, degrading the fidelity of the 3D reconstruction. Since our approach is only dependent on the quality of 2D keypoints, it shows slightly degraded performance only when the noise is injected over the input 2D keypoints. Qualitative 3D reconstruction performance of our approach in Fig. 5.1.1, 5.2 shows the visual improvement over the classical multi-view triangulation approaches when operated over noisy 2D keypoints.

5.1.2 Generation to unseen 2D data

There exists now several multi-view approaches for 3D human pose estimation that leverage either full or weak 3D supervision [18, 20, 24, 38, 39, 40, 46]. None of these references, however, directly tackle the unsupervised multi-view 3D reconstruction problem and hence are not as general as our solution. However, to showcase the generalization capability of our approach, we include these approaches in our evaluation, shown in Tab. 5.4. Furthermore, we also compare against recent monocular unsupervised 3D reconstruction methods. We leverage the processed datasets by

Method	**Detected 2D**	**GT 2D**
Iskakov et al. [18]	20.8	-
Remelli et al.[39]	30.2	-
Kadkhodamohammadi et al. [20]	49.1	-
Tome et al. [46]	52.8	-
Pavlakos et al. [38]	56.9	-
Multi-view Martinez [33]	57.0	-
Rhodin et al.[40]	51.6	-
Kocabas et al [24]	45.04	-
Kocabas et al. (SS w/o R) [24]	70.67	-
PRN [37]	124.5	86.4
RepNet [47]	65.1	38.2
Iqbal et al.[17]	69.1	-
Pose-GAN [28]	173.2	130.9
Deep NRSfM [26]	-	104.2
C3DPO [35]	153.0	95.6
MV NRSfM (Ours)	**45.2**	**30.2**

Table 5.4: Generalization experiments. Red tint rows have 3D supervision. Green tint are unsupervised 3D reconstruction methods. Our method is on par with most 3D supervised methods, and outperforms all unsupervised methods.

Dovotny et al. [35] as the detected 2D keypoints for a fair evaluation. We use the evaluation split of H3.6M dataset for this comparison. We find that our approach clearly outperforms all other unsupervised approaches, and is on-par with many supervised methods.

Chapter 6

Discussions

This thesis proposes a multi-view NRSfM architecture that incorporates neural shape prior using the recent advances of modern deep learning methods. We observe that two-physical views achieve comparable fidelity to complex, expensive setups that use multi-view triangulation. We also show the generalization capability of the proposed approach by generating accurate 3D reconstructions on unseen data. Although we require two rigid views at any instant of time, our approach still requires multiple non-rigid atemporal views to enforce the proposed neural shape prior. Literature in the domain of neural shape priors is extensive [35, 48] and new innovations are proposed constantly, and we believe we could leverage these innovations within our framework as one of the parts of our future direction.

6.1 Future Directions

6.1.1 Spatio-temporal neural prior

The proposed neural shape prior is able to help bring the regularization within the neural network. On top of bringing the spatial neural prior, one immediate step we propose is to bring in the temporal information to bring about the regularization within our network. A spatio-temporal neural prior could be leveraged to reason about both the modalities of information simultaneously.

6.1.2 Implicit pooling

As shown in the appendix, although the proposed network used max-pooling in the architecture, it is robust to different pooling operations in the fact that the network is agnostic to this operation. With this observation, one potential idea could be the investigation into implicit feature selection from multiple views instead of the max-pooling of the features currently used in our architecture.

Appendix A

Appendix

A.1 Reproducibility details

The Kronecker product in Eq. (6) increases the implementation complexity of our approach. To eliminate it and make parameter sharing easier in modern deep learning environments (e.g. TensorFlow, PyTorch), we reshape the filters and features and show that the matrix multiplication in each step of the encoder and decoder can be equivalently computed via a multi-channel convolution $*$ and transposed convolution $*^\top$ *i.e.* first layer on Eq. (6) could be implemented as

$$({}^1\mathbf{D}^{\#})^\top \cdot \mathbf{W}_k^{(n)} \implies {}^1\mathbf{d}^{\#} *^\top \mathbf{w}^{(n)}$$

where ${}^1\mathbf{d}^{\#} \in \mathbb{R}^{3\times 1\times {}^1B\times P}$ and $\mathbf{w}^{(n)} \in \mathbb{R}^{1\times 2\times P}$. Here, the filter dimension is height $\times$ width $\times \#$ of in-channel $\times \#$ of out-channel, while the feature dimension is height $\times$ width $\times \#$ of channel. Similarly, the subsequent layer operation is carried out as

$$({}^2\mathbf{D} \otimes \mathbf{I}_3)^\top \cdot {}^1\boldsymbol{\Psi}_k^{(n)} \implies {}^2\mathbf{d}^{\#} *^\top {}^1\Psi_k^{(n)}$$

where ${}^2\mathbf{d}^{\#} \in \mathbb{R}^{1\times 1\times {}^2B\times {}^1B}$ and ${}^1\Psi_k^{(n)} \in \mathbb{R}^{3\times 2\times {}^1B}$. Following the similar operations, the layers on the decoder part in Eq. (8) could be written as

$${}^L\mathbf{D} \cdot {}^L\boldsymbol{\psi}^{(n)} \implies {}^L\mathbf{d} * {}^L\psi^{(n)}$$

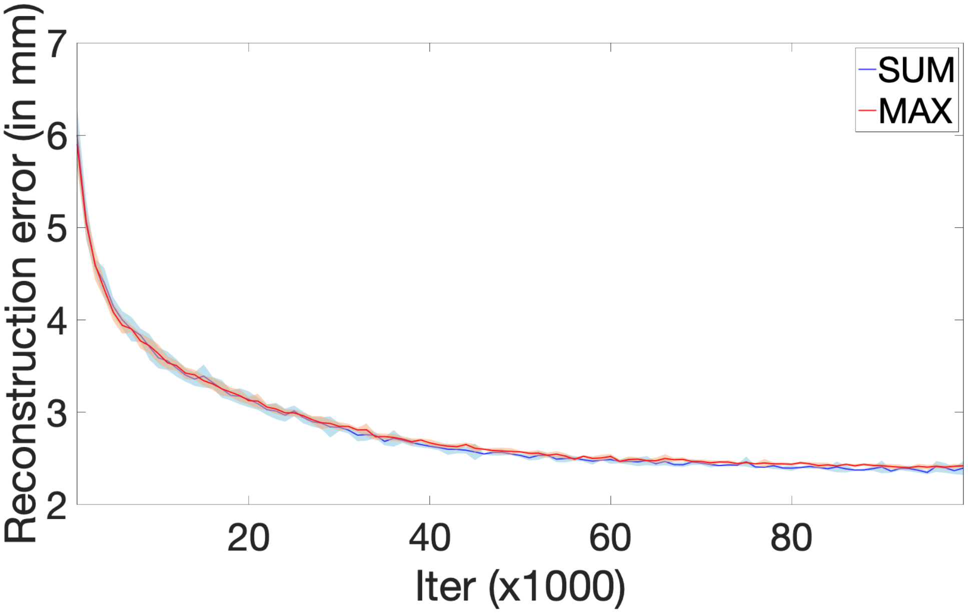

Figure A.1: 3D reconstruction error in MPJPE with different pooling operations. For each configuration, MV NRSfM is run 5 times and visualized with average accuracy (solid lines) together with standard deviation (shaded areas).

where ${}^{L}\mathbf{d} \in \mathbb{R}^{1\times1\times\,{}^{L}B\times\,{}^{L-1}B}$ and ${}^{L}\boldsymbol{\psi}^{(n)} \in \mathbb{R}^{1\times1\times\,{}^{L-1}B}$.

A.2 Robustness against pooling operation

As shown in the figure below, we run our approach with different pooling operations, *i.e.* max and sum pooling on human hands dataset. To account for the stochastic behavior due to network initialization and gradient descent, we run the approach 5 times and visualize with average accuracy (solid lines) together with standard deviation (shaded areas). MV NRSfM gives similar results for either pooling operation. This indicates our method is robust to the type of pooling operation that combines features from multiple views. As part of future avenues, we propose to investigate an implicit formulation to choose among the features coming from different views as part of the pooling operation.

A.3 Differentiable OnP solution

We find the solution to Orthographic-N-Point (OnP) problem for extracting the K rotation matrices from 2D-3D correspondences between K input poses $\mathbf{W}_k$ and canonicalized 3D structure $\mathbf{S}$, as shown in Fig. 2. We opt to use an algebraic solution that is computationally more light-weight compared to OnP solvers iteratively minimizing the geometric error. Although an algebraic solution does not necessarily reach local minima, it still leads to equivalent training performance of using a geometric solution. The benefit of using an algebraic solution is that it could be implemented as a differentiable operator, which could be easily accomplished via modern deep learning autograd packages. We choose a solution that finds the closed-form least square solution $\tilde{\mathbf{R}}^*$ for minimizing the reprojection error shown in Eq. (9), by subsequently projecting the $\tilde{\mathbf{R}}^*$ to become a rotation matrix $\mathbf{R}^* \in SO(3)$ using SVD. Due to its differentiability, we could easily insert the solution directly within our pipeline.

An alternative solution to find $\mathbf{R}$ is using an approximation from the network that we keep at the Rotation Factorization (RF) layer as a fully connected layer connecting ${}^{L}\mathbf{\Psi}_k^{(n)}$ and ${}^{L}\boldsymbol{\psi}_k^{(n)}$ and a linear combination among each blocks of ${}^{L}\mathbf{\Psi}_k^{(n)}$ to estimate $\mathbf{R}_k^{(n)}$, where the fully connected layer parameters are learned from the data. Our closed-form solution $\mathbf{R}^*$ from solving the OnP problem now implicitly acts as supervisory signal for the $\mathbf{R}_k^{(n)}$ generated by the network explained above.

A.4 Triangulation baseline

To obtain a 3D structure for corresponding synchronized 2D keypoints as a baseline, we utilize the triangulation method as the one given in [14] by leveraging epipolar geometry. Kocabas et al. [24] provide an open-source implementation for this method. Iterative Linear Least Squares (Iterative-LS) or Iterative-Eigen method is utilized as our baseline. The idea of the iterative linear method is to change the weights of the linear equations adaptively so that the weighted equations correspond to the errors in the 2D coordinate measurements.

Bibliography

[1] Antonio Agudo, Melcior Pijoan, and Francesc Moreno-Noguer. Image collection pop-up: 3d reconstruction and clustering of rigid and non-rigid categories. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2607–2615, 2018. 2, 4

[2] Ijaz Akhter, Yaser Sheikh, and Sohaib Khan. In defense of orthonormality constraints for nonrigid structure from motion. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1534–1541. IEEE, 2009. 1.2, 2, 4

[3] Ijaz Akhter, Yaser Sheikh, Sohaib Khan, and Takeo Kanade. Nonrigid structure from motion in trajectory space. In *Advances in neural information processing systems*, pages 41–48, 2009. 2

[4] Praneet C Bala, Benjamin R Eisenreich, Seng Bum Michael Yoo, Benjamin Y Hayden, Hyun Soo Park, and Jan Zimmermann. Openmonkeystudio: automated markerless pose estimation in freely moving macaques. *bioRxiv*, 2020. (document), 2, 3.1, 5.1, 5.3, 5.1, 5.1.1

[5] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm with application to wavelet-based image deblurring. In *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 693–696. IEEE, 2009. 4.1.2

[6] Christoph Bregler, Aaron Hertzmann, and Henning Biermann. Recovering non-rigid 3d shape from image streams. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, volume 2, pages 690–696. IEEE, 2000. 1.2, 2, 3.2, 4

[7] Hristos S Courellis, Samuel U Nummela, Michael Metke, Geoffrey W Diehl, Robert Bussell, Gert Cauwenberghs, and Cory T Miller. Spatial encoding in primate hippocampus during free navigation. *PLoS biology*, 17(12):e3000546, 2019. 2

[8] Yuchao Dai, Hongdong Li, and Mingyi He. A simple prior-free method for non-rigid structure-from-motion factorization. *International Journal of Computer*

Vision, 107(2):101–122, 2014. 1.2, 2, 4

[9] Katerina Fragkiadaki, Marta Salas, Pablo Arbelaez, and Jitendra Malik. Grouping-based low-rank trajectory completion and 3d reconstruction. In *Advances in Neural Information Processing Systems*, pages 55–63, 2014. 1.2, 2, 4

[10] Richard A Gibbs, Jeffrey Rogers, et al. Evolutionary and biomedical insights from the rhesus macaque genome. *science*, 316(5822):222–234, 2007. 2

[11] Ralph Gross, Iain Matthews, Jeffrey Cohn, Takeo Kanade, and Simon Baker. Multi-pie. *Image and Vision Computing*, 28(5):807–813, 2010. 2, 3.1

[12] Darcy L Hannibal, Eliza Bliss-Moreau, Jessica Vandeleest, Brenda McCowan, and John Capitanio. Laboratory rhesus macaque social housing and social changes: implications for research. *American Journal of Primatology*, 79(1): e22528, 2017. 2

[13] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 3.1, 5.1.1

[14] Richard I Hartley and Peter Sturm. Triangulation. *Computer vision and image understanding*, 68(2):146–157, 1997. 5.1.1, A.4

[15] Brian Hrolenok, Tucker Balch, David Byrd, Rebecca Roberts, Chanho Kim, James M Rehg, Scott Gilliland, and Kim Wallen. Use of position tracking to infer social structure in rhesus macaques. In *Proceedings of the Fifth International Conference on Animal-Computer Interaction*, pages 1–5, 2018. 2

[16] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1325–1339, 2013. 2, 5.1

[17] Umar Iqbal, Pavlo Molchanov, and Jan Kautz. Weakly-supervised 3d human pose learning via multi-view images in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5243–5252, 2020. 3.3

[18] Karim Iskakov, Egor Burkov, Victor Lempitsky, and Yury Malkov. Learnable triangulation of human pose. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7718–7727, 2019. 3.1, 3.3, 5.1.2

[19] Hanbyul Joo, Hao Liu, Lei Tan, Lin Gui, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh. Panoptic studio: A massively multiview system for social motion capture. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3334–3342, 2015. 2, 2, 3.1

[20] Abdolrahim Kadkhodamohammadi and Nicolas Padoy. A generalizable approach

for multi-view 3d human pose regression. *Machine Vision and Applications*, 32 (1):1–14, 2021. 3.1, 3.3, 5.1.2

[21] Sinead Kearney, Wenbin Li, Martin Parsons, Kwang In Kim, and Darren Cosker. Rgbd-dog: Predicting canine pose from rgbd sensors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8336–8345, 2020. 2

[22] Matt J Kessler, John D Berard, and Richard G Rawlins. Effect of tetanus toxoid inoculation on mortality in the cayo santiago macaque population. *American journal of primatology*, 15(2):93–101, 1988. 2

[23] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5.1

[24] Muhammed Kocabas, Salih Karagoz, and Emre Akbas. Self-supervised learning of 3d human pose using multi-view geometry. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1077–1086, 2019. 3.1, 3.3, 5.1.1, 5.1.2, A.4

[25] Chen Kong and Simon Lucey. Prior-less compressible structure from motion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4123–4131, 2016. 2, 4

[26] Chen Kong and Simon Lucey. Deep non-rigid structure from motion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1558–1567, 2019. 1, 2, 3.2, 3.3, 4, 4, 4.1.1, 5.1

[27] Chen Kong, Rui Zhu, Hamed Kiani, and Simon Lucey. Structure from category: a generic and prior-less approach. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 296–304. IEEE, 2016. 2, 4

[28] Yasunori Kudo, Keisuke Ogaki, Yusuke Matsui, and Yuri Odagiri. Unsupervised adversarial learning of 3d human pose from 2d joint locations. *arXiv preprint arXiv:1803.08244*, 2018. 3.3

[29] Suryansh Kumar. Non-rigid structure from motion: Prior-free factorization method revisited. In *Winter Conference on Applications of Computer Vision (WACV 2020)*, 2020. 2

[30] Suryansh Kumar. Non-rigid structure from motion: Prior-free factorization method revisited. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 51–60, 2020. 1.2, 2, 4

[31] Suryansh Kumar, Yuchao Dai, and Hongdong Li. Multi-body non-rigid structure-from-motion. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 148–156. IEEE, 2016. 2, 4

[32] Seong Hun Lee and Javier Civera. Closed-form optimal two-view triangula-

tion based on angular errors. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2681–2689, 2019. 5.1.1

[33] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2640–2649, 2017. 3.3

[34] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*, pages 483–499. Springer, 2016. 2

[35] David Novotny, Nikhila Ravi, Benjamin Graham, Natalia Neverova, and Andrea Vedaldi. C3dpo: Canonical 3d pose networks for non-rigid structure from motion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7688–7697, 2019. 3.2, 3.3, 4, 5.1.2, 6

[36] Vardan Papyan, Yaniv Romano, and Michael Elad. Convolutional neural networks analyzed via convolutional sparse coding. *The Journal of Machine Learning Research*, 18(1):2887–2938, 2017. 4.1.2, 4.1.2

[37] Sungheon Park, Minsik Lee, and Nojun Kwak. Procrustean regression networks: Learning 3d structure of non-rigid objects from 2d annotations. In *European Conference on Computer Vision*, pages 1–18. Springer, 2020. 3.3

[38] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. Harvesting multiple views for marker-less 3d human pose annotations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6988–6997, 2017. 3.1, 3.3, 5.1.2

[39] Edoardo Remelli, Shangchen Han, Sina Honari, Pascal Fua, and Robert Wang. Lightweight multi-view 3d pose estimation through camera-disentangled representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6040–6049, 2020. 3.1, 3.3, 5.1.2

[40] Helge Rhodin, Jörg Spörri, Isinsu Katircioglu, Victor Constantin, Frédéric Meyer, Erich Müller, Mathieu Salzmann, and Pascal Fua. Learning monocular 3d human pose estimation from multi-view images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8437–8446, 2018. 3.1, 3.3, 5.1.2

[41] Terence Sim, Simon Baker, and Maan Bsat. The cmu pose, illumination, and expression (pie) database. In *Proceedings of Fifth IEEE International Conference on Automatic Face Gesture Recognition*, pages 53–58. IEEE, 2002. 2, 3.1

[42] Carsten Steger. Algorithms for the orthographic-n-point problem. *Journal of Mathematical Imaging and Vision*, 60(2):246–266, 2018. (document), 4, 4.1

[43] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution rep-

resentation learning for human pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5693–5703, 2019. 2

[44] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 529–545, 2018. 2

[45] Carlo Tomasi and Takeo Kanade. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, 9(2):137–154, 1992. 1.2

[46] Denis Tome, Matteo Toso, Lourdes Agapito, and Chris Russell. Rethinking pose in 3d: Multi-stage refinement and recovery for markerless motion capture. In *2018 international conference on 3D vision (3DV)*, pages 474–483. IEEE, 2018. 3.1, 3.3, 5.1.2

[47] Bastian Wandt and Bodo Rosenhahn. Repnet: Weakly supervised training of an adversarial reprojection network for 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7782–7791, 2019. 3.3

[48] Chaoyang Wang, Chen-Hsuan Lin, and Simon Lucey. Deep nrsfm++: Towards 3d reconstruction in the wild. *arXiv preprint arXiv:2001.10090*, 2020. 6

[49] Xiaowei Zhou, Menglong Zhu, Spyridon Leonardos, Konstantinos G Derpanis, and Kostas Daniilidis. Sparseness meets deepness: 3d human pose estimation from monocular video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4966–4975, 2016. 2, 4

[50] Yingying Zhu, Dong Huang, Fernando De La Torre, and Simon Lucey. Complex non-rigid motion 3d reconstruction by union of subspaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1542–1549, 2014. 2, 4

[51] Christian Zimmermann, Duygu Ceylan, Jimei Yang, Bryan Russell, Max Argus, and Thomas Brox. Freihand: A dataset for markerless capture of hand pose and shape from single rgb images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 813–822, 2019. (document), 5.1, 5.1