FOUND: <u>F</u>oot <u>O</u>ptimization with <u>U</u>ncertain <u>N</u>ormals for Surface <u>D</u>eformation Using Synthetic Data

Oliver Boyne   Gwangbin Bae   James Charles   Roberto Cipolla

Department of Engineering, University of Cambridge, U.K.

{ob312, gb585, jjc75, rc10001}@cam.ac.uk

Abstract

Surface reconstruction from multi-view images is a challenging task, with solutions often requiring a large number of sampled images with high overlap. We seek to develop a method for few-view reconstruction, for the case of the human foot. To solve this task, we must extract rich geometric cues from RGB images, before carefully fusing them into a final 3D object. Our FOUND approach tackles this, with 4 main contributions: (i) SynFoot, a synthetic dataset of 50,000 photorealistic foot images, paired with ground truth surface normals and keypoints; (ii) an uncertainty-aware surface normal predictor trained on our synthetic dataset; (iii) an optimization scheme for fitting a generative foot model to a series of images; and (iv) a benchmark dataset of calibrated images and high resolution ground truth geometry. We show that our normal predictor outperforms all off-the-shelf equivalents significantly on real images, and our optimization scheme outperforms state-of-the-art photogrammetry pipelines, especially for a few-view setting. We release our synthetic dataset and baseline 3D scans to the research community.
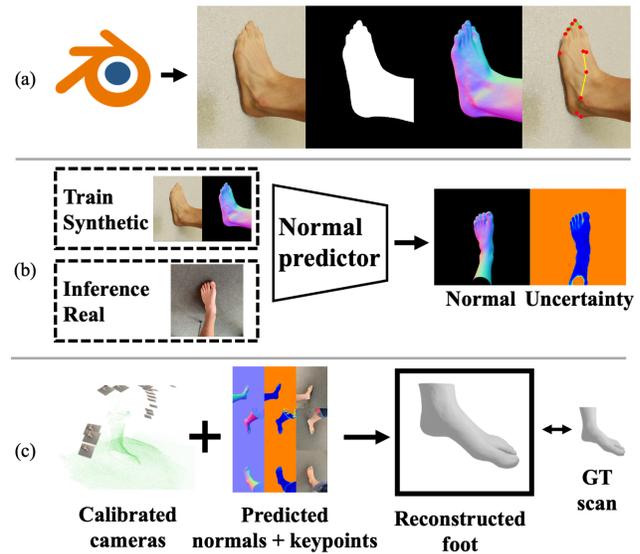
Figure 1. Method overview: (a) we use Blender [17] to synthetically render foot images, masks, surface normals, and keypoints; (b) we train a normal predictor on this data; (c) we predict normals on real images and optimize in a multi-view, calibrated setup to reconstruct the foot, evaluating on a ground truth scan.

1. Introduction

3D reconstruction of human body parts from images is a challenging computer vision task, of significant interest to the health, fashion and fitness industry. In this paper, we address the problem of human foot reconstruction. Shoe retail, orthotics, and personal health monitoring all benefit from accurate models of the foot, and the growth of the digital market for these industries has made the prospect of recovering a 3D foot model from images very appealing.

Existing solutions for foot reconstruction fit into one of four categories: (i) expensive scanning equipment [2, 6, 7]; (ii) reconstruction of noisy point clouds, *e.g.* from depth maps [30] or phone-based sensors such as a *TrueDepth* camera [8]; (iii) Structure from Motion (SfM) followed by Multi-View Stereo (MVS) [34, 35]; and (iv) fitting generative foot models to image silhouettes [26].

We find none of these solutions are satisfactory for accurate scanning in a home setting: expensive scanning equipment is not accessible to most consumers; phone-based sensors are limited in availability and ease of use, and noisy point clouds are difficult to use for downstream tasks such as rendering and taking measurements; SfM is dependent on a large number of input views to match dense features between images, and MVS can also produce noisy point clouds; and (until recently) foot generative models have been low quality and restrictive, and using only silhouettes from images limits the amount of geometrical information that can be obtained from the images, especially problematic in a few-view setting. The performance of these methods is also limited by the lack of paired images and 3D ground truth data for feet available for training.

To this end, we introduce FOUND, **F**oot **O**ptimisation using **U**ncertain **N**ormals for surface **D**eformation - which seeks to improve on typical multi-view reconstruction optimization schemes by leveraging per-pixel surface normals along with uncertainties. Similar to [26], our method only requires a small number of calibrated RGB images as input. While [26] relied only on silhouettes (which lack geometric information), we exploit additional cues - surface normals and keypoints. To address the lack of data, we also release a large-scale dataset of photorealistic synthetic images paired with ground truth labels for such cues. We outline our key contributions as follows:

- To facilitate research on 3D foot reconstruction, we release *SynFoot*, a **large-scale synthetic dataset** of 50,000 photorealistic foot images, coupled with accurate silhouette, surface normal, and keypoint labels. While collecting such data for real images requires expensive scanning equipment, our dataset is highly scalable. Despite only containing 8 real world foot scans, we show that our synthetic dataset sufficiently captures variation within foot images enough for downstream tasks to generalize to real images. We also release an **evaluative dataset** of 474 images of 14 real feet, each paired with ground truth per-pixel surface normals and a high-resolution 3D scan. Finally, we release our custom Python library for generating large scale synthetic datasets efficiently using Blender [17].

- We demonstrate that an **uncertainty-aware surface normal estimation network**, trained purely on our synthetic data from 8 foot scans, generalizes to real in-the-wild foot images. We use aggressive appearance and perspective augmentation to close the domain gap between synthetic and real foot images. The network estimates per-pixel surface normals and corresponding uncertainty. The uncertainty is useful in two regards: we can get accurate silhouettes for free (*i.e.* without training a separate network) by thresholding the uncertainty; and we can use the estimated uncertainty to weight the surface normal loss in our optimization scheme, increasing robustness against potential inaccuracy of the predictions made in certain views.

- We propose an **optimization scheme** capable of fitting a generative foot model [13] to a set of calibrated images with predicted surface normals and keypoints via differentiable rendering. Our pipeline is uncertainty-aware, capable of reconstructing a watertight mesh from very few views, can be applied to data collected from a consumer mobile phone, and beats state-of-the-art photogrammetry for surface reconstruction.

2. Related work

Synthetic dataset generation. In recent years, the capabilities for rendering photorealistic images in large quantities has improved significantly. This has resulted in a growing interest in generating high quality synthetic datasets for computer vision tasks. Synthetic data has the benefits of being much more scalable, cheap and accurate than typical data collection pipelines - especially for tasks difficult or impossible for human labellers. Many works have shown that synthetic data can be used as the primary source of training data for various deep learning problems. For human body reconstruction tasks, synthetic data has been used for bodies [36], faces [11,37], and eyes [38].

Single-image surface normal estimation. The goal of this problem is to estimate the pixel-wise surface normal, defined in the camera reference frame. While classical approaches [20, 27] relied on handcrafted image features and simplified the problem by discretizing the output and solving a classification problem, recent methods [10,18,28] use deep neural networks to directly regress the per-pixel normal. Unlike depth, surface normals are not affected by scale ambiguity and can be estimated from low-level cues like edges and shading [10], improving generalization. We follow the approach by Bae *et al.* [10] to quantify the uncertainty associated with the normal predictions and use it to improve the robustness of our foot optimization.

Normal integration. A surface normal map can be integrated to recover 3D shape up to a scale ambiguity. Methods like [15,16] optimize per-pixel depth such that the normal map computed from the depth map is consistent with the input normal map. However, such methods (1) cannot recover the scale, (2) are not applicable for multi-view images, (3) are unable to reconstruct unseen parts of the object and (4) are not uncertainty-aware (i.e. sensitive to inaccurate normal predictions). Conversely, we optimize the foot shape based on multi-view uncertainty-aware normal and keypoint constraints, so that the contribution of high-uncertainty predictions can be down-weighted. The optimized foot shape is both water-tight and accurately scaled, useful for commercial applications like virtual shoe try-on.

Human foot reconstruction. Human foot reconstruction is a task of interest to the footwear and orthotics industries.

Some foot reconstruction methods seek to reconstruct point clouds of feet [30], often leveraging LiDAR or structured light sensors available in some modern mobile phones. These sensors are not ubiquitous however, and these often noisy point cloud reconstructions are limited in their use for downstream tasks, often useful only to take a small number of measurements of the foot.

Other methods fit a generative model to predicted image silhouettes [26]. Building statistically-derived generative models for feet is difficult, in part because there are no large scale datasets available to the research community, outside of population measurement statistics [21, 22]. While generative models of the foot that use Principal Component Analysis (PCA) are not new [9], these have been very limited in their resolution and capabilities until recently. Recent work by Osman *et al.* [32] introduced SUPR, a PCA model of the human foot to be combined with the SMPL [29] full body model for the task of expressive, full body reconstruction. Boyne *et al.* [13] also recently released a generative model of the human foot, FIND. Instead of PCA, this model uses an implicit network to define per-vertex deformations to deform a template mesh into a target pose and shape. A dataset of high resolution foot scans was released with FIND.

Multi-view reconstruction. Reconstruction of depth and geometry from images requires known camera positions and internal parameters. These can be obtained directly from a capturing device using Augmented Reality (AR) technologies and Inertial Measurement Units (IMUs), or via a sparse 3D reconstruction from Structure From Motion (SfM) [34].

From this, a common method for reconstructing surface geometry is Multi-View Stereo (MVS) [14, 35], which involves the fusion of features matched across views. To produce a mesh as output, a surface reconstruction algorithm is required [12, 23].

Recent work has ventured into using neural rendering for surface reconstructions [31, 39], in which some neural representation of a 3D scene is learned which matches the reference views when rendered. Such methods are often dependent on a high number of input views and require large amounts of training time to produce a reconstruction.

3. Method

3.1. SynFoot - Synthetic dataset

We generate 50,000 images at 480×640 resolution using Blender's [17] Cycles engine. We replicate the behavior of images typically used for foot reconstruction commercial applications: captured from a handheld device, in a vertical arc around the foot, with the foot placed on the floor. We generate this dataset using a custom Python library built for large scale synthetic dataset rendering, which we release to the research community.

Camera. We sample camera positions in a vertical arc above the foot, uniformly sampling a radius from 30 to $40\,\mathrm{cm}$, angle to the vertical from -0.4π to 0.4π rad, and lateral displacement from 0 to $10\,\mathrm{cm}$. We use a camera focal length of $30\,\mathrm{mm}$, and point the camera at the world origin.

Figure 2. Samples from *SynFoot*, our synthetic dataset. We show (a) RGB, (b) silhouettes, (c) surface normals, and (d) keypoints. Further examples included in the supplementary.

Leg models. We find current generative models lack necessary photorealism, and do not capture enough geometry up the leg, for use in our synthetic dataset. As a result, we use 8 foot and leg scans adapted from the Foot3D dataset [13] - we find that a small number of leg scans, coupled with a large variety in environment and viewpoint, provide sufficient training data to train networks on downstream tasks. We also align one of two trouser models (with a randomly selected texture) to the model.

As the scan data does not capture the entire leg, there exists a horizontal cutoff plane on each mesh. From most views this is not visible, but certain viewpoints would suffer reduced realism as a result. We avoid this by (a) covering the plane with trousers, and (b) avoiding viewpoints where we detect that the cutoff plane would occupy more than 20% of the rendered image.

Lighting. We sample one of 14 HDRI environments [5], with a random intensity value to simulate variable lighting conditions. We add a point light at a random horizontal position, to add further realistic lighting variation. We also

add a plane behind the camera, with dimensions of a typical mobile phone, to replicate the shadow typically seen in mobile phone captured images.

Floor surface. We sample one of 34 textures collected from a number of license-free asset stores [1, 3–5] to model variety in flooring. We use diffuse, normal, reflectance, and roughness maps where available to provide realism to the surface. To replicate specular reflections seen on real surfaces, we apply a uniformly sampled [0, 0.3] specularity value to 20% of the dataset samples.

Compositing. As shown in Figure 2, for each sample we render: **RGB** images; **silhouettes** of the visible leg scan; **surface normals** relative to the camera, normalized from [-1, 1] to [0, 255], with XYZ corresponding to RGB; and **keypoints** based on 12 handlabelled keypoints on each scan (detailed in Figure 3).

3.2. Label prediction

So that we can optimize a generative foot model to real images, we seek to use our synthetic dataset to learn to predict useful representations of real data for subsequent fitting: per-pixel surface normals, silhouettes, and keypoints.

Surface normal prediction. Following [10], we predict the surface normal *probability distribution* and train the network by minimizing the negative log-likelihood (NLL) of the ground truth. Given N pixels (indexed with i) with ground truth surface normal $\mathbf{n}_i^{\text{gt}}$, the loss can be written as

$$
\begin{aligned}
\mathcal{L}_{\text{norm}}^{\text{train}} &= -\frac{1}{N} \sum_i^N \log p_{\text{AngMF},i}(\mathbf{n}_i^{\text{gt}}|\boldsymbol{\mu}_i, \kappa_i) \\
&= \frac{1}{N} \sum_i^N \kappa_i \cos^{-1} \boldsymbol{\mu}_i^T \mathbf{n}_i^{\text{gt}} + \log \frac{(1 + \exp(-\kappa_i \pi))}{(\kappa_i^2 + 1)},
\end{aligned}
\tag{1}
$$

where $\boldsymbol{\mu}_i$ and κ_i are the predicted mean direction and the concentration parameter. The objective of the training is to minimize the angle between $\boldsymbol{\mu}_i$ and $\mathbf{n}_i^{\text{gt}}$ while reducing κ_i for the pixels with high error. The expected angular error can then be used to quantify the uncertainty [10].

For our synthetic dataset, the ground truth is provided only for the foot. To encourage the network to estimate high uncertainty for the background, we generate noisy labels for the background pixels by sampling them uniformly on a unit hemisphere facing the camera. For the background pixels, the same loss (Equation 1) is computed and is downweighted by a factor 0.1. When back-propagating the loss for the background pixels, the gradient is detached for $\boldsymbol{\mu}$ to only influence the uncertainty.

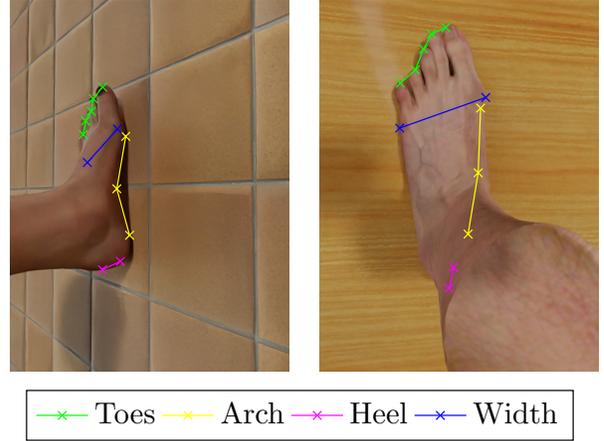

—×— Toes —×— Arch —×— Heel —×— Width

Figure 3. 12 keypoints defined for our dataset on two synthetic images - detailed keypoint definitions included in the supplementary.

Data augmentation. While our synthetic images look realistic, there still remains substantial domain gap with respect to the in-the-wild real images of human foot. Inspired by [11], we add aggressive data augmentation to close this domain gap. Specifically, we add random horizontal flipping, JPEG compression and Gaussian blur/noise, together with randomization of brightness, contrast, saturation and hue. We also add perspective augmentation by rotating the camera around the yaw, pitch and roll axes. More detail is provided in the supplementary material.

Silhouette prediction. From the surface normal prediction, we can get the expected value of the angular error. We conveniently obtain silhouettes by masking out the pixels with uncertainty higher than $30°$.

Keypoint prediction. Similar to surface normal prediction, we estimate the probability distribution for the keypoint locations, parameterized with a 2D Gaussian with a diagonal covariance matrix. We train a ResNet-50 [19] model to predict for an image, for each of the 12 keypoints visualized in Figure 3: its 2D image position $\mathbf{k}$ normalized to unit image coordinates $\bar{\mathbf{k}}$; a visibility flag v (whether within the image bounds); and predicted uncertainty $\boldsymbol{\sigma} = (\sigma_x, \sigma_y)$. We learn to predict keypoints and uncertainty by comparing against ground truth keypoints $\mathbf{k}^{\text{gt}}$,

$$
\begin{aligned}
\mathcal{L}_{\text{kp}}^{\text{train}} = \frac{1}{N} \sum_i^N \hat{v}_i \bigg(& \left\| \begin{pmatrix} (\bar{\mathbf{k}}_{i,j,x} - \bar{\mathbf{k}}_{i,j,x}^{\text{gt}})/\bar{\sigma}_{x,i} \\ (\bar{\mathbf{k}}_{i,j,y} - \bar{\mathbf{k}}_{i,j,y}^{\text{gt}})/\bar{\sigma}_{y,i} \end{pmatrix} \right\|_2^2 \\
& + \log \bar{\sigma}_{x,i}^2 \bar{\sigma}_{y,i}^2 \bigg)
\end{aligned}
\tag{2}
$$

We learn visibilities with an ℓ_2 loss comparing to ground

truth visibility. Similarly to our normal network, we train on our 50,000 synthetic images, adding affine and pixel color augmentations to the input data to improve the quality of training. We include examples showing the performance of the keypoint predictor on real images in the supplementary.

3.3. Multi-view fusion

Once we can predict surface normals, silhouettes and keypoints for real input images, we seek to fit a generative foot model to the images.

Generative model. For few-view fitting, a generative model is critical to enforce a strong prior over the foot space, and reduce the dimensionality of the optimization task. We use FIND [13], an implicit-explicit model which captures shape, pose and texture. The model is composed of a template mesh and an MLP F. The MLP predicts, for given shape and pose encodings z_s and z_p, the deformation for a template vertex at position x to deform it to the target shape,

$$\Delta x = F(x, z_s, z_p) \tag{3}$$

The application of F to every point x on the template results in a deformed, explicit mesh. This is then transformed by registration parameters r, t, and s - Euler rotation, translation, and XYZ scaling respectively. The FIND model is also capable of producing per-vertex colours parameterized by a texture encoding - in this work, we fix this encoding, as our focus is on accurate geometry reconstruction.

Differentiable rendering. Using PyTorch3D [33], we differentiably render surface normals and silhouettes in the camera reference frame, and project keypoints onto the image, for optimization.

3.4. Optimization

We now have the tools to optimize the FIND model to the predicted labels. We optimize the registration (alignment) parameters $\{r, t, s\}$ and shape and pose parameters $\{z_p, z_s\}$. We optimize over N views, each of size (H, W).

Keypoint loss. For K keypoints, we compute the ℓ_2 distance between the ith keypoint projected in the jth view and normalized to unit coordinates $\bar{\mathbf{k}}_{i,j}$ and that predicted by our keypoint predictor $\hat{\bar{\mathbf{k}}}_{i,j}$. We ignore 'invisible' keypoints using the visibility flag $\hat{v}$, and use our predicted normalized uncertainty $(\bar{\sigma}_{x,i}, \bar{\sigma}_{y,i})$,

$$\mathcal{L}_{\text{kp}}^{\text{opt}} = \frac{1}{NK} \sum_{i}^{N} \sum_{j}^{K} \hat{v}_{i,j} \left\| \begin{pmatrix} (\bar{\mathbf{k}}_{i,j,x} - \hat{\bar{\mathbf{k}}}_{i,j,x})/\bar{\sigma}_{x,i} \\ (\bar{\mathbf{k}}_{i,j,y} - \hat{\bar{\mathbf{k}}}_{i,j,y})/\bar{\sigma}_{y,i} \end{pmatrix} \right\|_2^2 \tag{4}$$

We use σ to down-weight the error for keypoints with high uncertainty - this typically corresponds to keypoints occluded in the image.

Normal loss. Given the estimated per-pixel surface normal probability distribution, parameterized by $\boldsymbol{\mu}$ and κ, we minimize the negative log-likelihood of the surface normal of the rendered foot model. The loss can be written as,

$$\mathcal{L}_{\text{norm}}^{\text{opt}} = \frac{1}{N} \sum_{i}^{N} \kappa_i \cos^{-1} \boldsymbol{\mu}_i^T \mathbf{n}_i^{\text{render}}. \tag{5}$$

Equation 5 differs from 1 in that the term only dependent on κ is discarded as it is not being updated. Equation 5 reduces the angular error between the predicted normal and the current estimate of the foot shape, while downweighting the error for predictions with high uncertainty (i.e. low κ).

Silhouette loss. An ℓ_2 loss between the rendered FIND model silhouette, and the pseudo-GT silhouette (found by thresholding the predicted normal uncertainty at $30°$).

3.5. Implementation details

Label prediction. We train the keypoint and surface normal predictors on our synthetic dataset. We use the Adam optimizer [25] for both, at a learning rate of 0.0001. We train the normal predictor for 20 epochs, and the keypoint predictor for 70.

Optimization scheme. We first downsample our images to increase speed and reduce memory usage. We optimize in two stages: (i) *registration*, with only FIND's $\{r, s, t\}$ as free parameters, and only a keypoint loss; and (ii) *deformation*, with free parameters $\{r, s, t, z_s, z_p\}$, and keypoint, silhouette and normal losses present.

We use the Adam optimizer, with a learning rate of 0.001, for 250 epochs and 1000 epochs respectively for the two stages. On a Titan X GPU, this takes approximately 2 minutes for 3 input views. We show in Figure 4 a visualization of the result of a reconstruction.

Where we restrict views for experimentation, we choose the restricted views by selecting views distributed evenly in the y direction (left-to-right as the images were taken).

Evaluative dataset. As described in Section 3.6, we use camera extrinsic and intrinsic parameters obtained using Structure from Motion in COLMAP [34]. This method requires a certain amount of overlap between views. We use these cameras to provide a fair comparison between our method and COLMAP, only measuring the reconstruction performance and not the quality of the camera calibration, and so that we can evaluate on our ground truth dataset.
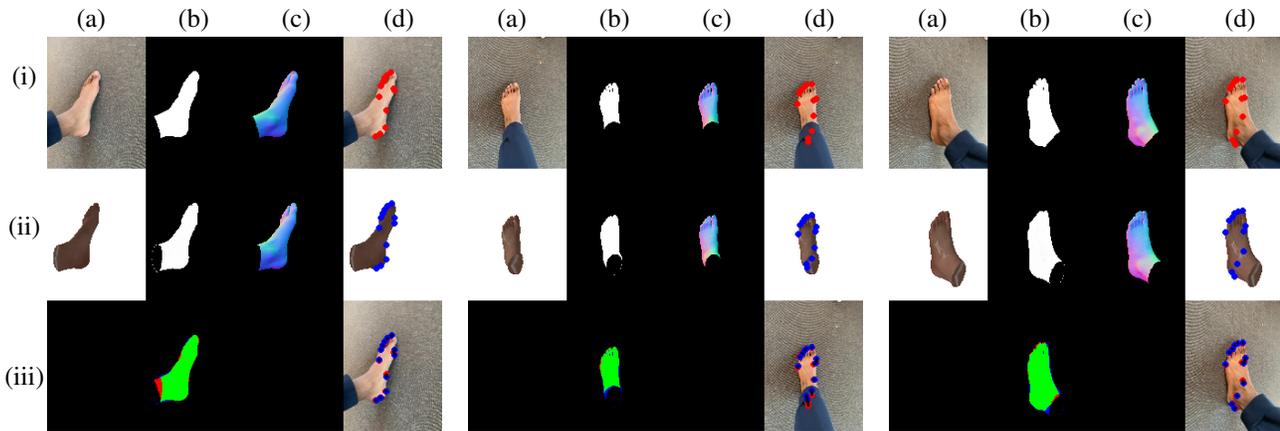
Figure 4. Here, we show our method optimizing to input images of a real foot, visualizing for three views: (a) RGB, (b) silhouette, (c) normals, and (d) keypoints, for (i) real input image, (ii) fitted FIND model, and (iii) the error. Note that the RGB for FIND uses FIND's default texture. To view the reconstruction quality compared to COLMAP and the GT scan, see Figure 8.
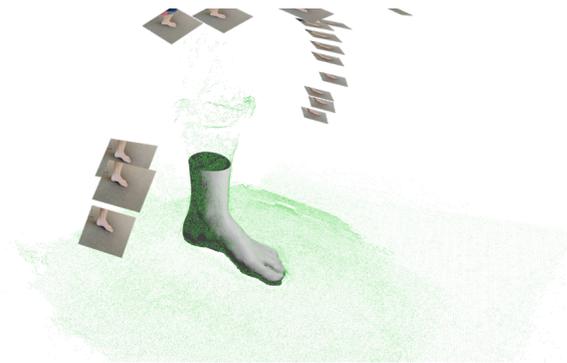

Figure 5. An example of a scan from our dataset. We show a sample of the calibrated images; the ground truth scan in grey; and the COLMAP reconstructed dense point cloud in green.

In practice, these parameters could be obtained directly by using tracking systems on mobile devices (*e.g.* ARKit on iOS). We show in Section 6 that our method can reconstruct successfully with cameras obtained via such approaches.

3.6. Experiments

Dataset. To evaluate multi-view reconstruction performance, we collect a dataset of calibrated image captures around static feet, alongsde 3D ground truth scans from an Artec Leo [2] scanner.

We perform sparse reconstruction via COLMAP [34], sufficient for relative camera alignment, with a scale ambiguity. While this would be sufficient for shape fitting (albeit with a scale ambiguity), to be able to evaluate our normal predictor and our fitting process, it is necessary to align the COLMAP result to the ground truth scan.

To do this, we perform dense reconstruction in COLMAP [35]. We run outlier detection on this recon-

structed dense point cloud, and floor detection and removal [40] on both the Leo scan and the COLMAP point cloud. We rotate and translate both so that the (now removed) floors are horizontal and at a height of zero. To align the two precisely, we run an optimization scheme to minimize the chamfer loss between the dense point cloud and the 3D scan, optimizing over four parameters: in-plane rotation, in-plane translation, and isotropic scale. We also use some outlier rejection stages to improve the accuracy of this process. An example output of this process can be seen in Figure 5.

Once this process is complete, we also conveniently can render the ground truth meshes onto each captured image to get ground truth surface normals, examples of which are in Figure 6. Note that this process was incredibly labour intensive, and produces a far smaller number of paired RGB and surface normal images than in our synthetic dataset.

The completed dataset contains 14 scenes of scanned feet, with a total of 474 calibrated images.

Normal predictor evaluation protocol. The accuracy of our surface normal estimation network is compared against COLMAP [35] and Bae *et al.* [10]. 20 images are sampled from each scan to form a test set of 280 images. Pixel-wise angular error is computed and the mean, median and root-mean-squared errors are reported. We also report the percentage of pixels with error less than $[11.25°, 22.5°, 30°]$. As we focus on the surface reconstruction of the foot, we discard the prediction for legs and trousers by slicing the ground truth mesh at 20 cm above the floor, and evaluating only on the pixels below the cutoff.

Optimization evaluation protocol. We evaluate the quality of our 3D optimization method by comparing the output mesh to a ground truth scan from our dataset. We select a sample of 10,000 points from each mesh, sampling

RGB   GT   COLMAP [35]   Bae *et al.* [10]   Ours   Ours - uncertainty
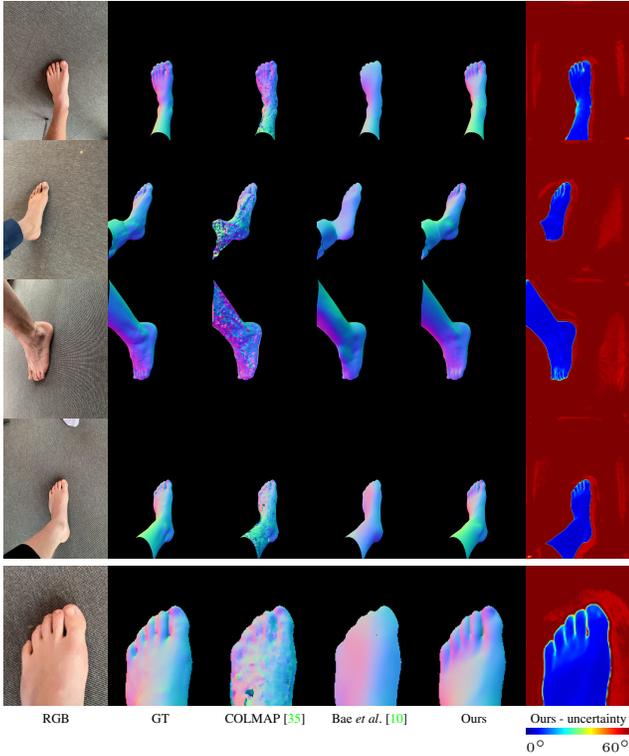
0°   60°

Figure 6. Qualitative comparisons of per-pixel normals, masking each prediction according to the GT. Further examples of in-the-wild surface normal predictions included in the supplementary.

Method	Normal error (°) ↓			Normal accuracy (%) ↑		
	Mean	Median	RMSE	11.25°	22.5°	30°
COLMAP [35]	28.24	17.22	40.88	33.00	60.46	70.33
Bae et al. [10]	28.54	25.89	32.79	12.43	41.27	59.75
Ours (aug. from [10])	25.50	19.66	32.26	27.02	55.66	67.61
Ours	**11.30**	**9.24**	**14.43**	**62.06**	**91.73**	**96.21**

Table 1. Evaluation metrics of normal predictions, comparing against our baseline GT scans.

uniformly over the surface area. We compare the nearest neighbor (NN) between each point in the two sampled point clouds, and evaluate two metrics: the NN distance, and the NN surface normal angular error.

We compare our method against a meshed COLMAP reconstruction, which is built from COLMAP's dense point cloud via Poisson surface reconstruction [24].

4. Results

Normal prediction. Table 1 compares the accuracy of our surface normal predictor to the baseline methods. We outperform both methods across all metrics. When our predictor is trained with data augmentation from [10], the performance degrades substantially, suggesting that the proposed

Method	NN chamfer error (μm) ↓			NN normal error (°) ↓		
	Mean	Median	RMSE	Mean	Median	RMSE
Ours	**15.2**	7.5	**29.3**	**12.9**	**9.9**	**17.4**
COLMAP	16.1	**3.3**	43.3	20.3	13.7	27.7

Table 2. 3D Results on fitting to all images available in each scan
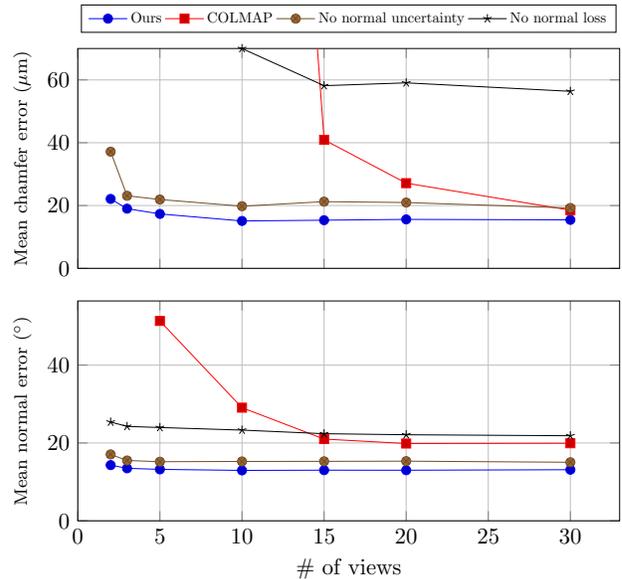

Figure 7. We show how the reconstruction quality of our method varies with number of input views, comparing with COLMAP, and with our method without normals and normal uncertainties. We show that COLMAP reconstruction fails for fewer than 15 views, whereas our method maintains accuracy for 3-5 views.

aggressive augmentation helps reduce the synthetic-to-real domain gap. Qualitative comparison in Figure 6 shows that our method captures the complex geometry around the toes.

3D fitting results. Table 2 shows the quantitative performance of our method compared to COLMAP for the many view case. Qualitative examples of the reconstruction quality can be seen in Figure 8. The results suggest that COLMAP can capture some geometry more precisely (seen by the lower median chamfer error), at the cost of a noisy reconstruction that has some larger errors, and poor surface normal consistency. Conversely, our method obtains a smoother reconstruction with fewer outliers, and surface normals that much more closely resemble the ground truth.

Number of views. We show in Figure 7 our method's ability to perform accurate reconstructions with few available views. COLMAP achieves similar reconstruction accuracy for 30+ views, but fails almost completely for fewer than 15 views. Our method, on the other hand, is able to

GT   COLMAP   Ours   GT   COLMAP   Ours

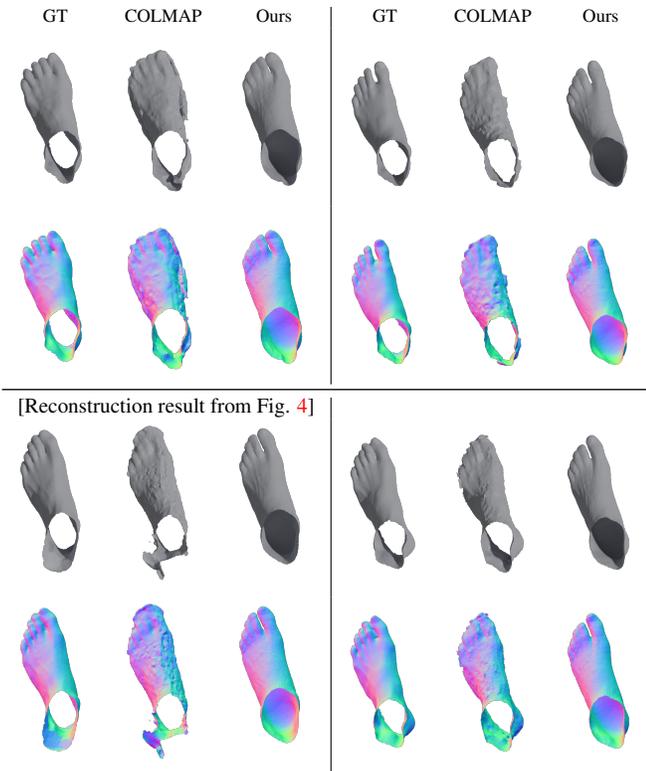[Reconstruction result from Fig. 4]

Figure 8. Qualitative results on four reconstructions from all available images (approximately 30 per scan). We show geometry and surface normal renders of the ground truth 3D mesh, the COLMAP reconstruction, and the results of our optimization method. Further examples included in the supplementary.

	NN Chamfer error (μm) $\downarrow$		NN Normal error ($^\circ$) $\downarrow$	
	3 view	20 view	3 view	20 view
Ours	**18.60**	**15.30**	**13.58**	**13.12**
w/o keypoint uncertainty	19.28	16.50	13.70	13.27
w/o normal uncertainty	23.08	21.05	15.57	15.37
w/o normal loss	102.11	59.20	24.23	22.13

Table 3. Ablation study on our 3D optimization method.

retain virtually all accuracy for as few as 3 views, and even handles the 2 view case with reasonable accuracy (better than COLMAP with 15). Furthermore, the graph shows that the use of our normal loss both substantially improves the reconstruction quality, and significantly improves our method's ability to handle few views.

Ablation study. We show results of our ablation of our optimization process in Table 3. The data shows that, for both the few view and many view cases, our normal loss is crucial for accurate surface reconstruction. Furthermore, using normal uncertainty, and to a lesser extent keypoint uncertainty, further improve the quality of reconstruction.
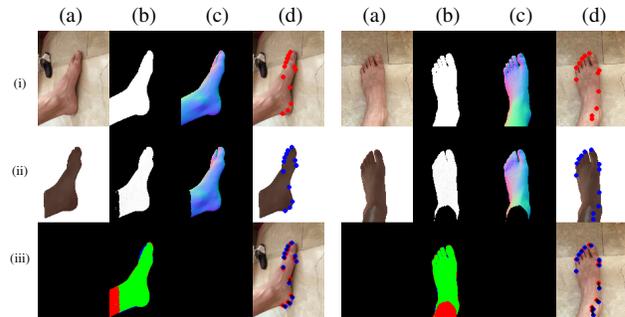

Figure 9. We show qualitative reconstruction on cameras obtained via AR tracking. We show here two out of the three used views, visualizing (a) RGB, (b) silhouette, (c) normals, and (d) keypoints, for (i) input image, (ii) fitted FIND model, and (iii) the error.

5. Conclusion

In this paper, we introduced our synthetic dataset *Syn-Foot*, that effectively captures variation in foot images despite only containing 8 real world leg scans. We have developed a method for learning to predict normals with uncertainty purely from this synthetic data, using aggressive data augmentation to bridge the synthetic-to-real domain gap and obtain accurate surface normal predictions on real images. Experimental results show that this method significantly outperforms off-the-shelf normal predictors, and MVS reconstruction. We have also shown a method for reconstructing a foot model from a small number of views that utilises the predicted normals with uncertainties, that outperforms typical MVS pipelines while requiring an order of magnitude fewer input images.

6. Future work

Generalization. While we only show results on the constrained case of a foot placed on the floor, nothing inherent in our method restricts us to this use case. As future work, we would like to extend this to posed feet in arbitrary positions, and to other body parts, such as hands.

Cameras from AR tracking. Our evaluation dataset shows results on cameras obtained via SfM, reliant on many views with significant overlap for accuracy. However, our method is also possible using camera parameters obtained via AR camera tracking. Although we do not have GT meshes, we show qualitatively on a private dataset with AR calibrated cameras that our optimization is able to fit to real images successfully. Figure 9 shows one example of this, and further examples are included in the supplementary.

7. Acknowledgements

References

[1] ambientCG, Public Domain Resources for Physically Based Rendering. `https://ambientcg.com`. Accessed: 2023-08-22. 4

[2] Artec Leo, Artec3D. `https://www.artec3d.com/portable-3d-scanners/artec-leo`. Accessed: 2023-08-20. 1, 6

[3] cgbookcase. `https://www.cgbookcase.com`. Accessed: 2023-08-22. 4

[4] Poliigon. `https://www.poliigon.com`. Accessed: 2023-08-22. 4

[5] Poly Haven. `https://polyhaven.com`. Accessed: 2023-08-22. 3, 4

[6] Volumental foot scanner. `https://volumental.com/volumental-retail-scanner`. Accessed: 2023-08-20. 1

[7] Vorum *YETI 3D* foot scanner. `https://vorum.com/yeti-3d-foot-scanner/`. Accessed: 2023-08-20. 1

[8] Xesto Inc. `https://www.xesto.io/`. Accessed: 2023-08-20. 1

[9] Edmée Amstutz, Tomoaki Teshima, Makoto Kimura, Masaaki Mochimaru, and Hideo Saito. PCA based 3D shape reconstruction of human foot using multiple viewpoint cameras. In *Computer Vision Systems: 6th International Conference, ICVS 2008 Santorini, Greece, May 12-15, 2008 Proceedings 6*, pages 161–170. Springer, 2008. 3

[10] Gwangbin Bae, Ignas Budvytis, and Roberto Cipolla. Estimating and exploiting the aleatoric uncertainty in surface normal estimation. In *Proc. of IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2, 4, 6, 7

[11] Gwangbin Bae, Martin de La Gorce, Tadas Baltrušaitis, Charlie Hewitt, Dong Chen, Julien Valentin, Roberto Cipolla, and Jingjing Shen. DigiFace-1M: 1 million digital face images for face recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3526–3535, 2023. 2, 4

[12] Fausto Bernardini, Joshua Mittleman, Holly Rushmeier, Cláudio Silva, and Gabriel Taubin. The ball-pivoting algorithm for surface reconstruction. *IEEE transactions on visualization and computer graphics*, 5(4):349–359, 1999. 3

[13] Oliver Boyne, James Charles, and Roberto Cipolla. FIND: An unsupervised implicit 3D model of articulated human feet. In *British Machine Vision Conference (BMVC)*, 2022. 2, 3, 5

[14] Neill DF Campbell, George Vogiatzis, Carlos Hernández, and Roberto Cipolla. Using multiple hypotheses to improve depth-maps for multi-view stereo. In *Computer Vision–ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part I 10*, pages 766–779. Springer, 2008. 3

[15] Xu Cao, Hiroaki Santo, Boxin Shi, Fumio Okura, and Yasuyuki Matsushita. Bilateral normal integration. In *European conference on computer vision (ECCV)*, 2022. 2

[16] Xu Cao, Boxin Shi, Fumio Okura, and Yasuyuki Matsushita. Normal integration via inverse plane fitting with minimum point-to-plane distance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2382–2391, 2021. 2

[17] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. 1, 2, 3

[18] Tien Do, Khiem Vuong, Stergios I Roumeliotis, and Hyun Soo Park. Surface normal estimation of tilted images via spatial rectifier. In *Proc. of European Conference on Computer Vision (ECCV)*, 2020. 2

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

[20] Derek Hoiem, Alexei A Efros, and Martial Hebert. Recovering surface layout from an image. *International Journal of Computer Vision*, 75:151–172, 2007. 2

[21] Ales Jurca, Tomaz Kolsek, and Tina Vidic. Dorothy mass foot measurement campaign. *Footwear Science*, 3(sup1):S83–S85, 2011. 3

[22] Ales Jurca, Jure Žabkar, and Sašo Džeroski. Analysis of 1.2 million foot scans from North America, Europe and Asia. *Scientific reports*, 9(1):1–10, 2019. 3

[23] Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, volume 7, page 0, 2006. 3

[24] Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Transactions on Graphics (ToG)*, 32(3):1–13, 2013. 7

[25] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[26] Felix Kok, James Charles, and Roberto Cipolla. FootNet: An efficient convolutional network for multiview 3D foot reconstruction. In *Proceedings of the Asian Conference on Computer Vision*, 2020. 1, 2, 3

[27] Lubor Ladicky, Bernhard Zeisl, and Marc Pollefeys. Discriminatively trained dense surface normal estimation. In *Proc. of European Conference on Computer Vision (ECCV)*, 2014. 2

[28] Shuai Liao, Efstratios Gavves, and Cees GM Snoek. Spherical regression: Learning viewpoints, surface normals and 3d rotations on n-spheres. In *Proc. of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[29] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, Oct. 2015. 3

[30] Nolan Lunscher and John Zelek. Point cloud completion of foot shape from a single depth map for fit matching using deep learning view synthesis. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 2300–2305, 2017. 1, 2

[31] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 3

[32] Ahmed A A Osman, Timo Bolkart, Dimitrios Tzionas, and Michael J. Black. SUPR: A sparse unified part-based human body model. In *European Conference on Computer Vision (ECCV)*, 2022. 3

[33] Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3D deep learning with PyTorch3D. *arXiv:2007.08501*, 2020. 5

[34] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 1, 3, 5, 6

[35] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*, pages 501–518. Springer, 2016. 1, 3, 6, 7

[36] Gul Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning from synthetic humans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2

[37] Erroll Wood, Tadas Baltrušaitis, Charlie Hewitt, Sebastian Dziadzio, Thomas J Cashman, and Jamie Shotton. Fake it till you make it: face analysis in the wild using synthetic data alone. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3681–3691, 2021. 2

[38] Erroll Wood, Tadas Baltrusaitis, Xucong Zhang, Yusuke Sugano, Peter Robinson, and Andreas Bulling. Rendering of eyes for eye-shape registration and gaze estimation. In *Proceedings of the IEEE international conference on computer vision*, pages 3756–3764, 2015. 2

[39] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. *Advances in Neural Information Processing Systems*, 33:2492–2502, 2020. 3

[40] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3D: A modern library for 3D data processing. *arXiv:1801.09847*, 2018. 6