Patch-based reconstruction of a textureless deformable 3D surface from a single RGB image

Aggeliki Tsoli and Antonis. A. Argyros
Foundation for Research and Technology - Hellas (FORTH)
{aggeliki, argyros}@ics.forth.gr

Abstract

We propose a deep learning method for reconstructing a textureless deformable 3D surface from a single RGB image, under various lighting conditions. One of the challenges when training a neural network to predict the shape of a deformable object is that the object exhibits such a great deal of shape variation that it is essentially impractical to have a training set consisting of all possible deformations the object may realize. However, different areas of the deformable object may exhibit similar types of deformations, e.g. similar wrinkles might appear in different areas on the surface of a cloth. Motivated by this, we propose learning local models of shape variation from image patches that we then combine into a global reconstruction of the observed object. Initially, we divide the input image into overlapping patches and a zero-mean depth map as well as a normal map are estimated for each patch using deep learning. Stitching of depth maps is performed by finding the optimal translation of each patch depth map along the viewing direction of the camera and averaging the depth predictions of neighboring patches at their overlapping areas. Stitching of normal maps is performed by normalizing and averaging the normals predictions of neighboring patches at their overlapping areas. Finally, bilateral filtering is performed on the stitched depth and normal maps in order to perform fine-scale smoothing at the regions around patch boundaries. We show increased accuracy compared to previous work even in the presence of limited training data and more effective generalization to unseen objects.

1. Introduction

Reconstructing objects in 3D from visual data has been a long standing problem in computer vision. We are particularly interested in reconstructing deformable objects that exhibit complex deformations, as this may have a number of potential applications in virtual reality and computer graphics. Deep learning approaches have shown impressive performance in a great range of computer vision tasks such as semantic segmentation, 3D object pose estimation, 3D scene reconstruction and more. Recently, deep learning has also been applied to reconstructing deformable objects from a single RGB image with promising results, both for textured [24, 9] as well as for textureless objects [6, 3].

A key attribute of deep learning methods is that their performance is strongly correlated with the amount of available training data. For instance, in order to predict the shape of a deformable object, the dataset used for training should ideally capture all possible variations in shape across the surface of the object. Modeling all possible deformations at every single area of the object's surface considering all areas simultaneously is impractical. However, there are many cases where objects exhibit similar deformations at different areas of their surface. For instance, a wrinkle at the bottom of a T-shirt may look similar to a wrinkle in the middle of a T-shirt. We pose that taking into account variations in local shape regardless of the location on the object's surface is an effective solution for dealing with the high-dimensional deformation space of deformable objects.

In this paper, we are inspired by previous work on building local deformation models [29], patch-based 3D reconstruction [8] and 3D reconstruction of textureless deformable objects from a single image [3] in order to provide accurate patch-based 3D reconstruction of a textureless 3D surface under various lighting conditions from a single RGB image. More specifically, we use deep learning to learn models of local 3D shape variation given an image patch, fit these local models on overlapping patches of the input image and stitch the resulting local geometries into a continuous deformable surface after estimating the relative translation among the depth maps using non-linear least-squares optimization. Learning local deformation models has multiple benefits. First, we utilize the data in our training set more effectively, thus, we need less training data for a desired level of reconstruction accuracy. Second, learning a local deformation model amounts to learning a simpler function than learning a global deformation model. Third, we are able to generalize more effectively to reconstruct-

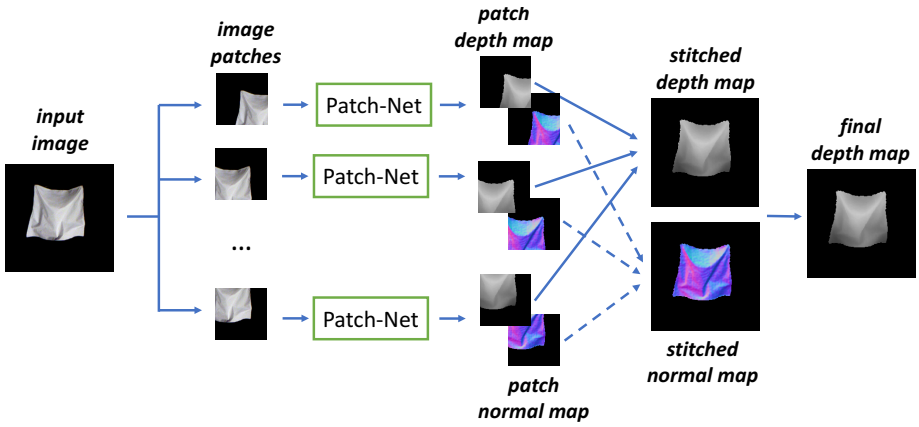

Figure 1: Patch-based 3D reconstruction from a single image. The input image is split into overlapping patches and prediction of depth and normal maps per patch is performed using Patch-Net. Patch depth maps are stitched by translating each depth map along the camera viewing direction in order to minimize the relative depth difference at overlapping regions and averaging the depth at overlapping regions. Patch normal maps are stitched by normalizing and averaging the predictions at overlapping regions. Both the stitched depth and normal maps are refined using bilateral filtering at the patch boundaries. The final depth map is estimated after normals integration with a depth scale factor inferred by the stitched depth map.

ing objects that are not in the training set, which is a very common case in real-life scenarios.

To the best of our knowledge, we present the first patch-based method for reconstructing deformable 3D surfaces employing deformation models for the patches that were generated using deep learning. Our proposed approach inherits all the benefits of patch-based methods such as increased accuracy even in the presence of limited data and more effective generalization to unseen objects.

2. Previous work

We review patch-based deep learning methods and relevant work on 3D reconstruction of deformable objects.

Deep learning using image patches: Although there has been limited work on patch-based deep learning methods, image or feature patches have been used successfully in the context of neural networks for tasks such as high-resolution image classification [12], texture synthesis [20] and image-to-image translation [13]. Hou et al. [12] use patch-level convolutional neural networks to identify cancerous tissues and train a decision fusion model to aggregate patch-level predictions. Isola et al. [13] and Li et al. [20] use a generative adversarial network with a patch-based discriminator network that distinguishes real from fake patches considering image and feature patches, respectively. They also show that best accuracy is achieved with medium-sized patches. In our work, we perform pixel-level predictions per patch and subsequently fuse the predictions from different patches using a non-linear optimization framework.

Deformable object reconstruction from a single image: Reconstruction of a deformable 3D object from a single image has been performed using Shape-from-Template (SfT) methods. SfT methods reconstruct the 3D shape of an object from a single RGB image given a reference 3D object template. Reconstruction is performed by leveraging image correspondences between the observed image and the object's texture and employing a proper deformation model as a prior. Previous work has shown highly accurate reconstruction for various types of objects such as isometric [4] or elastic surfaces [11], deformable 3D objects [23] as well as poorly textured objects [32]. Lately, an SfT method based on deep learning, DeepSfT [15], was proposed. A neural network with an auto-encoder is trained to predict a normalized depth map of the object of interest and an image warp between the observed image relative to the reference image. A second auto-encoder refines these predictions generating the output depth map. DeepSfT is tailored to a specific object and, thus, it requires an extensive synthetic dataset with the deformations of each object of interest.

Deep learning-based methods for reconstructing deformable 3D objects from a single image have only recently being proposed. They reconstruct an object with predefined mesh topology [24] or number of vertices [9, 6] relying mostly on synthetic data. Pumarola et al. [24] predicts a rectangular 3D mesh of fixed topology from an image that is geometrically consistent up to Procrustes alignment with 3D ground truth data. Thus, the detail of the predicted deformations is inevitably limited by the resolution of the 3D mesh used for training, a 9×9 rectangular mesh. In Golyanik et al. [9], a 3D point cloud of fixed size is directly regressed from image data using an auto-encoder architec-

ture. Training and evaluation is performed only with synthetic data. DeepGarment [6] relies on Convolutional Neural Networks to learn the mapping from rendered garment images to 3D vertex displacements from a template mesh representing the underlying 3D garment model. Although the results are promising, the predicted 3D meshes exhibit rather coarse deformations.

Ground truth data from both synthetic and real sequences in the form of 3D meshes is hard to acquire. We bypass that by training directly on real data that can be easily captured using a commodity depth camera. Moreover, we are able to reconstruct objects with very different shape from the shape of objects used for training by learning local shape models.

Patch-based 3D reconstruction: Patch-based approaches for 3D reconstruction have been a long standing trend in the computer vision community as it has been shown that they provide more effective generalization to previously unknown objects and greater robustness to noise, object articulation and limited training sets compared to global approaches. In Shape-from-Template, local surface deformation models based on nonlinear Gaussian Process Latent Variable Models [29] as well as simpler linear models in conjunction with inextensibility constraints [28] are used for 3D surface reconstruction from a single image. In Structure from Motion (SfM), Fayad et al. [8] reconstruct a surface given a monocular video sequence by dividing the surface into overlapping patches, reconstructing each of these patches independently using a quadratic deformation model and finally registering them by constraining points shared by patches to be at the same 3D location. Russell et al. [27] segment the scene into a constellation of object parts, recognize parts that are likely to constitute objects and subsequently join them to reconstruct the scene. Haene et al. [10] present an energy formulation for depth map recovery from image data utilizing a patch-based prior and apply the proposed framework to depth map fusion and computational stereo. Kozlov et al. [19] propose an approach for 3D reconstruction and tracking of dynamic surfaces using a single depth sensor. Each input depth image is subdivided into non-rigidly connected surface patches, it is deformed towards the canonical pose by estimating a rigid transformation for each patch and then a surfel-based technique is employed for fusing the 3D reconstructions of the patches.

Shape-from-Shading: Shape-from-Shading is an inherently under-constrained problem and various assumptions about the lighting and the shape of the scene have been made in order to solve it. Durou et al. [7] present an extensive overview on early work on Shape-from-Shading. Latest approaches tend to infer jointly two or more modalities that contribute to the image formation process, such as albedo, depth, normals, reflectance and lighting parameters in an optimization-based [35, 2, 5] or learning-based [26, 14, 31, 30] manner.

Our approach is mostly related to the work presented in [3] and [34]. In [34], the authors assume diffuse Lambertian shading and infer from each image patch the distribution of quadratic surfaces that are likely to have produced it. We instead learn a mapping from an image patch to the corresponding 3D shape expressed as a depth map while being robust to illumination variation by training on data with various types of lighting. Similar to [3], we predict a depth map and a normal map from a single RGB image adopting an auto-encoder architecture using a shared encoder for depth and normals prediction. However, we employ a patch-based formulation for solving the same problem leading to increased reconstruction accuracy and generalization ability.

3. Method

3.1. Problem formulation

We aim at reconstructing a textureless deformable surface from a single RGB image. Let $\mathbf{I} \in \mathbb{R}^{H\times W\times 3}$ be an RGB image of size $W \times H$ and $\mathbf{M} \in \mathbb{R}^{H\times W}$ a binary mask that highlights the foreground region at $\mathbf{I}$. Our goal is, given the input image $\mathbf{I}$, to predict a depth map $D \in \mathbb{R}^{H\times W}$ corresponding to the 3D reconstruction of the observed foreground object inside the mask $\mathbf{M}$ and the corresponding normal map $N \in \mathbb{R}^{H\times W\times 3}$.

Our approach consists of three steps. Initially, we define a set of overlapping patches $\mathbf{P}_i \in \mathbb{R}^{H_p\times W_p\times 3}, i = 1,\ldots,\mathcal{P}$ of size $H_p \times W_p$ on the input image $\mathbf{I}$ and estimate a depth map D_i and normal map N_i for each image patch via deep learning (Section 3.2). We, subsequently, stitch the predicted depth maps D_i into a single depth map D' and normal map N' considering the overlapping pixels among patches (Section 3.3). Finally, we refine D', N' via bilateral filtering on the areas around patch boundaries. The result is the output depth map D denoting the 3D reconstruction of the observed foreground object and the corresponding normal map N (Section 3.4).

3.2. Patch shape prediction from RGB

To reconstruct the shape per patch, we follow the latest practices [3, 31] of predicting disentangled representations. More specifically, we use a neural network that we term Patch-Net, which is illustrated in Figure 2, where given an input image patch we predict the relative depth map and the normal map associated with that patch. Patch-Net is an autoencoder network with a single encoder branch and two decoder branches for depth and normals prediction. We use a 3×3 kernel in all 2D convolutional layers and stride of 1. The channels per layer vary from 64 to 512 as shown in Figure 2. In essence, Patch-Net is a simplified SegNet-style architecture where compared to previous work [1, 3] we drop the skip connections between encoder and decoder

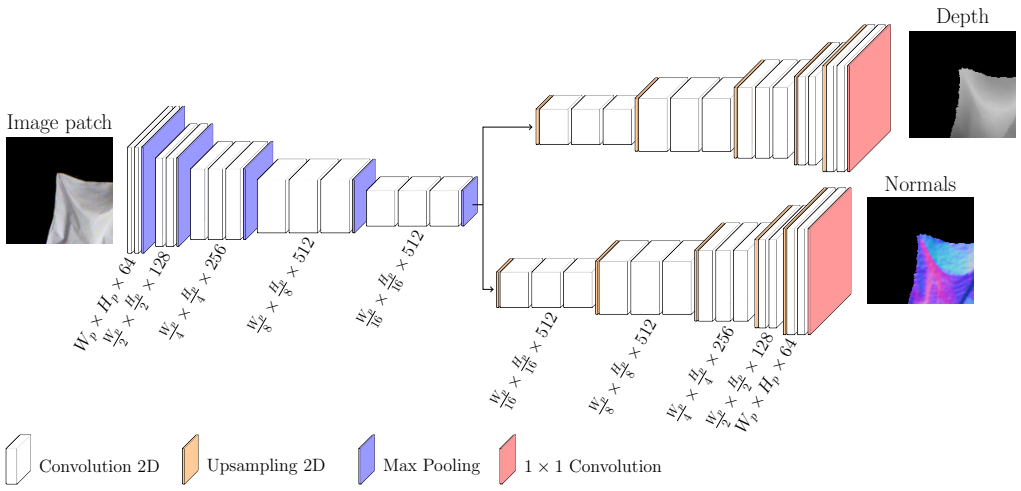

Figure 2: The Patch-Net architecture. An image patch is fed into an auto-encoder network that outputs the depth and normal map corresponding to the input patch. Depth and normals are predicted jointly using a shared encoder.

at the max unpooling layers and perform single stage training for estimating depth and normals. We also introduce an additional 1×1 convolutional layer for depth and normals prediction at the end of the decoder, use ReLU instead of linear activations and more importantly, we predict relative depth in the scene based on the mean depth per patch instead of absolute depth measurements taken directly from the captured RGB-D data [3], or depth normalized in $[-1, 1]$ [15]. That makes our predictions invariant to the location of the object in the scene while retaining the units of the object dimensions in the captured 3D scene.

To train our neural network, we define $\mathcal{P}$ patches of size $H_p \times W_p$ per input image in our training set spread uniformly across each image. That leads to a training set of patches where each sample $(\mathbf{P}^s, \mathbf{M}^s, \mathbf{D}^s, \mathbf{N}^s), s = 1, \ldots, S$ contains a patch $\mathbf{P}^s$ of an input RGB image, the foreground mask $\mathbf{M}^s$ corresponding to the patch, the relative ground truth depth map $\mathbf{D}^s$ and the ground truth normal map $\mathbf{N}^s$ of the patch. Let $\mathbf{Q}^s$ be the depth map recorded with an RGB-D sensor corresponding to patch $\mathbf{P}^s$. In order to be invariant to the absolute position of the object in our training data, we use the relative depth $\mathbf{D}^s$ per patch by subtracting the mean depth of the foreground pixels p of the patch:

$$\mathbf{D}^s = \mathbf{Q}^s / Z, \tag{1}$$

where

$$Z = \sum_p \mathbf{M}_p^s \mathbf{Q}_p^s / \sum_p \mathbf{M}_p^s. \tag{2}$$

Let $\Phi_D : \mathbb{R}^{H_p \times W_p \times 3} \rightarrow \mathbb{R}^{H_p \times W_p}$ be the function that predicts the depth map of a patch given an input image patch in Patch-Net and $\Phi_N : \mathbb{R}^{H_p \times W_p \times 3} \rightarrow \mathbb{R}^{H_p \times W_p \times 3}$ the function that predicts the normal map of the input image patch in Patch-Net. We train Patch-Net using standard losses for depth and normals prediction [3, 33, 18].

Depth loss: Training for depth prediction is performed by penalizing the absolute difference of the predicted depth map $\Phi_D(\mathbf{P}^s)$ for a patch from the ground truth depth map $\mathbf{D}^s$ of the same patch. Note that only the pixels p that correspond to the foreground, i.e. non-zero $\mathbf{M}_p^s$, actually contribute to the loss.

$$\mathcal{L}_D = \frac{1}{S} \sum_{s=1}^{S} \frac{\sum_p |\mathbf{D}^s - \Phi_D(\mathbf{P}^s)| \mathbf{M}_p^s}{\sum_p \mathbf{M}_p^s}. \tag{3}$$

Normals loss: Let $\hat{\mathbf{N}}_i^s = \Phi_N(\mathbf{P}^s)$ be the predicted patch normals. Training for normals prediction is performed via loss $\mathcal{L}_N$ using a linearized version of the cosine similarity [33] (loss $\mathcal{L}_a$) while favoring unit length normals (loss $\mathcal{L}_l$).

$$\mathcal{L}_N = \frac{1}{S} \sum_{s=1}^{S} \frac{\sum_p (\kappa \mathcal{L}_a(\mathbf{N}_i^s, \hat{\mathbf{N}}_i^s) + \mathcal{L}_l(\hat{\mathbf{N}}_i^s)) \mathbf{M}_p^s}{\sum_p \mathbf{M}_p^s} \tag{4}$$

where

$$\mathcal{L}_a(\mathbf{N}_i^s, \hat{\mathbf{N}}_i^s) = \arccos\left(\frac{\mathbf{N}_i^s \hat{\mathbf{N}}_i^s}{||\mathbf{N}_i^s|| ||\hat{\mathbf{N}}_i^s|| + \epsilon}\right) \frac{1}{\pi}, \tag{5}$$

$$\mathcal{L}_l(\hat{\mathbf{N}}_i^s)) = (||\hat{\mathbf{N}}_i^s|| - 1)^2. \tag{6}$$

We use $\epsilon = 10^{-6}$ to prohibit division by zero and $\kappa = 10$ as the relative weight between the two terms $\mathcal{L}_a, \mathcal{L}_l$ that constrain the prediction of normals.

Total loss: We train Patch-Net by weighing equally the depth and normals losses:

$$\mathcal{L} = \mathcal{L}_D + \mathcal{L}_N. \tag{7}$$

3.3. Patch stitching

Given a test image $\mathbf{I}$, we predict a depth map $D_i = \Phi_D(\mathbf{P}_i)$ and a normal map $N_i = \Phi_N(\mathbf{P}_i), i = 1, \ldots, \mathcal{P}$ per patch on image $\mathbf{I}$. Then, we stitch the depth maps and normal maps of all patches into a single depth map and a single normal map at the resolution of the test image, respectively.

3.3.1 Stitching depth maps of patches

In order to stitch the predicted relative depth map patches D_i into a unified depth map, we translate each reconstructed patch along the viewing direction of the camera so that overlapping areas between neighboring patches correspond to the same 3D points in the scene. More specifically, let Q_{ij} be the overlapping image region, expressed in terms of pixel indices k_i for depth map D_i and pixel indices k_j for depth map D_j, between two neighboring patches $\mathbf{P}_i, \mathbf{P}_j$. We apply a translation offset t_i at each patch depth map so that the distance between translated neighboring patches is minimized over their overlapping region:

$$t_i' = \arg\min_{t_i} \sum_{i=1}^{\mathcal{P}} \sum_{j=1}^{\mathcal{P}} \sum_{k_i, k_j \in Q_{ij}} ||D_i[k_i] + t_i - D_j[k_j] - t_j||_2^2. \tag{8}$$

After estimating the optimal translations t_i' along depth, the translated depth map per patch is $D_i' = D_i + t_i'$. To constrain the solution of the nonlinear optimization system above to a single solution, we set $t_1 = 0$ and optimize for the rest of the translations. Optimization is performed using the Levenberg Marquardt algorithm [21] as implemented in Python [1]. Finally, we stitch the translated patch depth maps D_i' into a single depth map D' corresponding to a reconstruction of the object in the original image $\mathbf{I}$ by averaging at each pixel location the depth predictions from the patches containing that pixel.

3.3.2 Stitching surface normal maps of patches

We stitch the normal maps $N_i, i = 1, \ldots, \mathcal{P}$ predicted by Patch-Net into a single normal map N' by normalizing and then averaging at each pixel location the predictions of normals from the patches containing that pixel.

3.4. Refinement of depth and normal maps

Bilateral filtering has been successfully used in the past for denoising depth maps while preserving object boundaries [22]. We apply bilateral filtering on D' at the regions around patch boundaries in order to smooth out small-scale discontinuities in depth and normals that were potentially not fully resolved during patch stitching while, at the same time, preventing foreground pixels close to the object's boundary from being influenced by background pixels. We do the same for N'. The outcome of this step is a refined depth map D and normal map N that correspond to the reconstruction of the deformable object from the input image $\mathbf{I}$.

[1] Function *scipy.optimize.leastsq*.

4. Experimental results

4.1. Overview

Evaluation datasets: We evaluate our method using various configurations of the dataset introduced in [3]. This dataset was obtained using a Kinect 1 depth sensor and contains RGB images, depth maps, normals maps and foreground masks of resolution 224×224 for 5 types of textureless deformable objects (cloth, T-shirt, sweater, hoody, paper) undergoing various types of deformations in various illumination settings with various light sources casting light from various locations and directions relative to the object. Because of the variability of the training dataset in illumination conditions, our proposed method does not require explicit modeling of lighting/shading. In total, the dataset contains 18 sequences of 15799 samples for cloth, 12 sequences of 6739 samples for T-shirt, 4 sequences of 2203 samples for sweater, 1 sequence of 517 samples for hoody and 3 sequences of 1187 samples for paper.

Evaluation metrics: We evaluate our method considering two types of errors; i.e, depth-based error (E_D) and angular error (E_A). The angular error E_A between predicted and ground truth normals is calculated using Eq. 5 and averaged for all foreground points. The depth-based error is calculated as the mean Euclidean distance between the 3D points of a pointcloud generated from a predicted depth map and the corresponding ground truth 3D points after Procrustes alignment [16], as in [24, 3]. We compare three ways of generating the predicted depth map, leading to the following variants of the depth-based error metric.

E_D^t : *Depth map generation via patch depth maps stitching.* We follow the approach described in Section 3.3.1 to get a stitched depth map D. However, because Patch-Net was trained with relative depths within a patch, D is located close to the origin. Thus, in order to calculate the depth-based error, we translate D along the viewing direction of the camera based on the difference between the mean predicted depth and the mean ground truth depth of the foreground object. D^t denotes the translated depth map.

E_D^N : *Depth map generation via patch normal maps stitching and normals integration.* It is known [25] that integrating normals under the assumption of perspective camera projection generates a depth map that is computable up to a multiplicative constant, i.e., up to scaling along the depth dimension. We examine the case of estimating the multi-

Experiment	Error	64	96	128	160	192	224
1. cloth-cloth	$E_D^t[mm]$	21.76 ± 6.64	17.25 ± 5.15	15.12 ± 4.73	13.56 ± 5.30	$\mathbf{11.95 \pm 4.41}$	12.61 ± 4.45
	$E_D^N[mm]$	21.97 ± 6.83	17.53 ± 5.22	14.74 ± 4.75	13.42 ± 5.30	$\mathbf{11.53 \pm 4.06}$	12.11 ± 4.36
	$E_D^{Ns}[mm]$	17.6 ± 4.92	13.30 ± 4.78	12.80 ± 4.45	12.92 ± 5.47	$\mathbf{10.89 \pm 3.85}$	11.57 ± 4.30
	$E_A[^o]$	19.37 ± 3.35	15.98 ± 4.51	14.72 ± 3.39	15.63 ± 4.18	$\mathbf{14.50 \pm 3.34}$	15.05 ± 3.89
2. tshirt-tshirt	$E_D^t[mm]$	19.82 ± 5.10	15.77 ± 3.92	14.83 ± 3.47	14.54 ± 4.22	$\mathbf{13.80 \pm 3.86}$	13.91 ± 3.80
	$E_D^N[mm]$	19.93 ± 5.30	15.94 ± 4.10	14.76 ± 3.76	14.39 ± 4.23	$\mathbf{12.94 \pm 3.69}$	13.38 ± 3.37
	$E_D^{Ns}[mm]$	15.76 ± 3.88	13.61 ± 3.66	13.70 ± 3.83	14.06 ± 4.24	$\mathbf{12.54 \pm 3.74}$	13.06 ± 3.39
	$E_A[^o]$	20.08 ± 3.84	18.67 ± 4.12	18.63 ± 4.43	20.15 ± 4.48	$\mathbf{19.41 \pm 4.45}$	19.51 ± 4.37
3. cloth-tshirt	$E_D^t[mm]$	25.97 ± 6.55	24.04 ± 6.08	24.03 ± 7.09	$\mathbf{22.23 \pm 6.12}$	24.77 ± 7.00	23.37 ± 5.91
	$E_D^N[mm]$	25.86 ± 6.87	24.14 ± 6.36	23.32 ± 6.71	$\mathbf{22.08 \pm 6.47}$	23.98 ± 6.67	23.19 ± 5.98
	$E_D^{Ns}[mm]$	23.35 ± 6.76	22.97 ± 6.42	22.74 ± 7.20	$\mathbf{22.03 \pm 6.92}$	23.49 ± 7.25	22.56 ± 6.76
	$E_A[^o]$	25.85 ± 3.38	24.83 ± 3.34	$\mathbf{24.29 \pm 3.80}$	25.09 ± 3.64	26.15 ± 4.07	25.94 ± 3.80
4. cloth-sweater	$E_D^t[mm]$	37.28 ± 8.96	32.22 ± 8.93	33.11 ± 9.45	$\mathbf{31.04 \pm 8.16}$	33.45 ± 9.90	39.61 ± 10.01
	$E_D^N[mm]$	36.62 ± 8.84	32.59 ± 8.62	32.45 ± 9.35	$\mathbf{28.63 \pm 7.33}$	32.60 ± 9.75	37.13 ± 9.80
	$E_D^{Ns}[mm]$	33.27 ± 8.40	30.08 ± 8.00	30.10 ± 10.00	$\mathbf{28.19 \pm 7.17}$	31.80 ± 9.44	35.89 ± 9.76
	$E_A[^o]$	30.19 ± 3.34	$\mathbf{27.74 \pm 3.74}$	27.94 ± 4.79	29.59 ± 4.56	30.96 ± 4.79	33.57 ± 5.61
5. cloth-hoody	$E_D^t[mm]$	37.17 ± 8.47	32.21 ± 7.18	$\mathbf{30.72 \pm 7.72}$	31.47 ± 9.35	33.48 ± 8.64	38.50 ± 10.33
	$E_D^N[mm]$	36.87 ± 8.69	32.26 ± 7.16	31.12 ± 8.29	$\mathbf{30.98 \pm 8.06}$	32.77 ± 8.11	37.91 ± 9.77
	$E_D^{Ns}[mm]$	36.58 ± 8.68	31.48 ± 7.03	31.09 ± 8.73	$\mathbf{30.24 \pm 8.16}$	32.59 ± 8.21	36.22 ± 9.34
	$E_A[^o]$	34.40 ± 2.85	32.98 ± 2.61	$\mathbf{29.73 \pm 2.52}$	31.34 ± 3.05	32.02 ± 3.08	33.49 ± 3.16
6. cloth-paper	$E_D^t[mm]$	17.98 ± 5.42	18.24 ± 4.83	$\mathbf{16.45 \pm 4.75}$	23.06 ± 5.89	21.95 ± 5.62	29.08 ± 5.70
	$E_D^N[mm]$	15.77 ± 4.89	15.98 ± 4.69	$\mathbf{15.10 \pm 4.44}$	20.41 ± 6.15	20.46 ± 5.81	27.00 ± 6.71
	$E_D^{Ns}[mm]$	15.55 ± 5.14	15.46 ± 4.86	$\mathbf{14.53 \pm 4.48}$	16.63 ± 5.06	17.12 ± 5.08	18.14 ± 5.30
	$E_A[^o]$	25.67 ± 5.80	26.35 ± 5.59	$\mathbf{24.52 \pm 5.96}$	31.22 ± 6.51	31.21 ± 5.58	40.45 ± 7.40

Table 1: Accuracy of the proposed method in reconstructing depth and normals considering various patch sizes ($64 - 224$). E_D^t, E_D^N, E_D^{Ns} correspond to the errors of different depth maps; each depth map was generated in a different way (see text).

plicative constant considering the depth map D^t generated using patch stitching and translation close to ground truth.

E_D^{Ns} : *Depth map generation via patch normal maps stitching and normals integration with known depth scaling.* Previous work [3] inferred depth scaling comparing the depth map computed based on normals integration, to the ground truth depth map. Extracting this information from ground truth improves the results, but also constitutes a limiting assumption. Our method does not require this assumption. However, for facilitating the comparison to [25], we show the increase in performance that this assumption would yield.

Evaluated methods: We evaluate our method against state-of-art on 3D reconstruction of textureless deformable objects from a single image [3]. Given that shading is an important cue for shape reconstruction, especially when the object of interest is textureless, we also compare against previous work on simultaneous inference of shape, illumination, and reflectance from shading [2] (SIRFS). To compare against [3], we used the implementation that is publicly available online[2]. To compare against [2], we use the error values reported in [3].

Implementation details: We have implemented our method in Keras using a Tensorflow backend. From each image in a training set, we have considered patches with overlap equal to half the patch size. In all experiments, training has been performed using an Adam optimizer [17] with a fixed learning rate of 0.001. We use a bilateral filter with kernel size 3×3 and $\sigma = 10cm$ for interpolating among the various depth values inside the kernel and $\sigma = 0.3$ for the normal maps.

[2] https://github.com/bednarikjan/texless_defsurf_recon

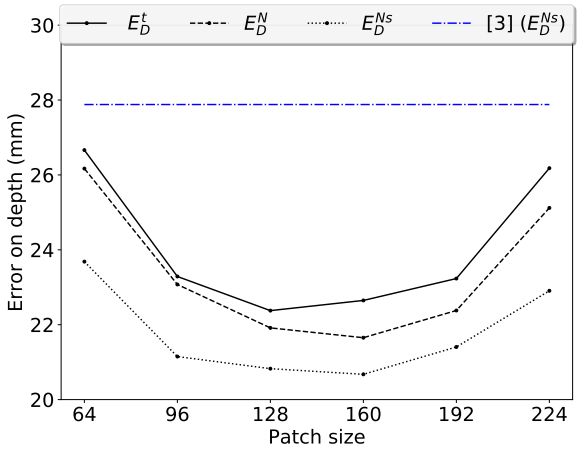

Figure 3: Aggregate error on depth prediction per patch size over all six experiments in [3]. E_D^t, E_D^N, E_D^{Ns} correspond to different depth maps, generated in different ways (see text). The blue line on top denotes the aggregate error of [3].

Experiment	SIRFS (E_D^{Ns})	Bednarik et al. [3] (E_D^{Ns})	Ours (E_D^{N})	OURS (E_D^{Ns})	SIRFS (E_A)	Bednarik et al. [3] (E_A)	Ours (E_A)
1. cloth-cloth	31.55 ± 10.93	17.53 ± 5.50	*14.74 ± 4.75*	$\mathbf{12.80 \pm 4.45}$	37.98 ± 23.18	17.37 ± 12.51	$\mathbf{14.72 \pm 3.39}$
2. tshirt-tshirt	31.09 ± 15.03	17.18 ± 18.58	*14.76 ± 3.76*	$\mathbf{13.70 \pm 3.83}$	30.17 ± 20.26	$\mathbf{18.07 \pm 12.71}$	18.63 ± 4.43
3. cloth-tshirt	30.29 ± 10.42	26.26 ± 7.72	*23.32 ± 6.71*	$\mathbf{22.74 \pm 7.20}$	30.08 ± 19.43	25.74 ± 15.81	$\mathbf{24.29 \pm 3.80}$
4. cloth-sweater	39.51 ± 14.96	38.93 ± 10.36	*32.45 ± 9.35*	$\mathbf{30.10 \pm 10.00}$	33.25 ± 21.60	31.52 ± 19.07	$\mathbf{27.94 \pm 4.79}$
5. cloth-hoody	43.51 ± 13.79	43.22 ± 24.81	*31.12 ± 8.29*	$\mathbf{31.09 \pm 8.73}$	36.84 ± 23.14	32.54 ± 21.15	$\mathbf{29.73 \pm 2.52}$
6. cloth-paper	49.35 ± 18.51	24.16 ± 7.15	*15.10 ± 4.44*	$\mathbf{14.53 \pm 4.48}$	56.69 ± 27.09	35.53 ± 22.16	$\mathbf{24.52 \pm 5.96}$

Table 2: Comparison with previous work on the datasets sets used in [3]. We report the error values for our method for patch size 128×128. Errors on depth (E_D^{Ns}, E_D^{N}) are expressed in mm and angular errors (E_A) in degrees.

4.2. Ablation study

Varying the patch size: We examine rectangular patches of width 64, 96, 128, 160, 196, 224 pixels (full image size) and evaluate our performance based on the six main experiments performed in [3]. We followed exactly the same experimental protocols regarding the definition of the training and test sets. Table 1 shows the reconstruction accuracy for each patch size as measured by the error metrics defined previously. We denote each experiment as X-Y where X is the object used for training and Y the object used for testing. For each experiment, we show in bold the result for the optimal patch size per error metric.

Considering the depth-based errors, when the training and testing objects are the same (experiments 1 and 2), the optimal patch size is relatively close to the full resolution of the image. Note, though, that in both experiments 1 and 2, testing is performed considering different directions of light compared to the training settings rather than different types of object deformations. For the rest of the experiments, we observe that performance increases while reducing the patch size, but up to a point. There seems to be a sweet spot around patch width equal to 128 upon which further decrease of the patch size actually leads to increased error as very small patches are not informative any more. In the last experiment, we observe that the error remains low for patches equal or smaller than 128×128. We believe that small patch sizes are favored because in this experiment training was performed on cloth deformations and testing on paper deformations that are way more coarse than the cloth deformations.

Figure 3 shows the aggregate error per patch size over all experiments for all depth-based error metrics. We remind that E_D^{Ns} denotes the idealized scenario employed in [3] where depth is predicted by integrating normals and using a depth scaling inferred from the ground truth. E_D^t and E_D^N show the error using the depth map generated via patch depth maps stitching and the error when integrating the stitched normals to generate a depth map inferring the depth scaling factor from the stitched depth map. As a reference, the blue line corresponds to the average error of the work in [3]. Regardless of the metric and the patch size, the proposed method outperforms [3] considerably. Based on Table 1, the angular error exhibits a similar behavior. Considering jointly the depth-based and angular errors, we deem the optimal patch size to be 128×128.

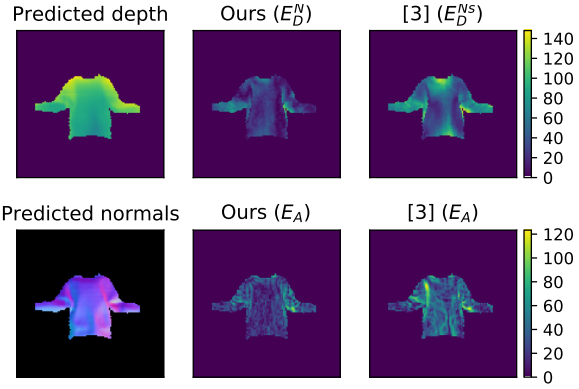

Figure 4: Prediction errors on a sample image in the sweater dataset after training on the cloth dataset. Left column: the predicted depth and normals with our method. Top row, last two columns: The error of the predicted depth in mm for our method and the method in [3]. Bottom row, last two columns: the error on the predicted normals in degrees for our method and [3].

Varying the overlap between neighboring patches: Increasing the overlap among neighboring image patches on the training images has the potential to increase the diversity of the considered patches. We claim that conceptually, this is similar to increasing the amount of training data. We refer the reader to the experiments provided in Section 4.3 with subsets of the original training sets of varying size.

4.3. Comparison with state-of-art

Evaluation on state-of-art datasets: In Table 2, we show the performance of our method with the selected optimal patch size 128×128 compared to SIRFS [2] and Bednarik et al. [3]. The results for SIRFS were obtained from [3]. The lowest error over all three methods is shown in bold.

Note that the results of previous work are shown after normals integration with depth scaling inferred from ground

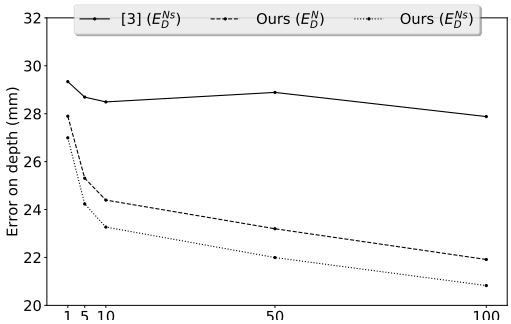

Figure 5: Performance of our method compared to [3] for training sets of size 1%, 5%, 10%, 50%, 100% relative to the size of the original training sets. We report the average depth prediction error over all six experiments in [3].

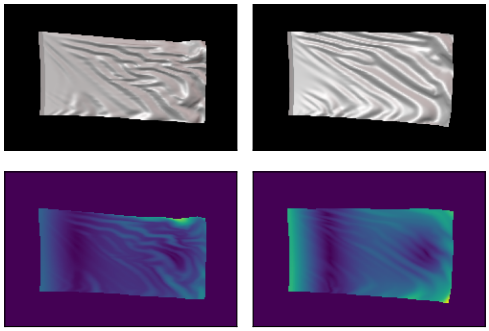

Figure 6: Example frames of a synthetic dataset with high-resolution images of a waving flag (top) and heatmaps of our error on depth reconstruction (bottom).

truth (E_D^{Ns}). We stress that even if we get rid of this limiting assumption (E_D^N), our method is still more accurate than previous work. The gain in performance using our patch-based method is most prominent when the training and testing objects vary greatly (experiments 4-6) showcasing the enhanced generalization ability of our proposed patch-based method. Figure 4 shows an example of the performance of our method and the method in [3] for the case of training on cloth deformations and testing on sweater deformations. The angular error is also overall lower in our proposed method.

Evaluation on varying training set size: We explore the reconstruction accuracy of our method for various training sets sizes compared to [3]. More specifically, we repeat the experiments performed in Section 4.3 taking into account random subsets of the initial training set of size 1%, 5%, 10% and 50%. We have generated 10 random subsets of size 1%, 8 subsets of size 5%, 5 subsets of size 10% and 2 subsets of size 50%. In Figure 5, we report the average error on depth prediction for each training set size considering all random subsets for the specific size and all six experiments. We observe that small training sets lead to less accurate predictions than large training sets for both methods, as expected. However, the proposed patch-based method leverages more effectively the available data as the size of the training set increases, thus, decreasing the reconstruction error at a faster rate than the global method in [3].

4.4. Qualitative results on high-resolution data

Currently, training deep learning architectures with high-resolution input images is computationally infeasible. Patch-based approaches like the proposed one, can work around this limitation by making predictions considering image patches and subsequently fusing the predictions per patch into a single prediction for the input high-res image [12]. We showcase the ability of our method to work with high-resolution input data within reasonable error limits using a synthetic sequence of resolution 1344×896 depicting a waving flag deformed by wind forces that vary in strength and direction over time (see Figure 6). The patch size was set to 224×224. We used 800 frames for training and 100 frames for testing with average error on depth 0.28 units for a flag of width equal to 4.9 units (error equals 5.7% of the flag's width) and angular error 16.02 degrees. Although we cannot directly compare the error on depth among experiments on synthetic and real data, the angular error is similar to the angular error in earlier experiments when training and testing on the same object. Qualitatively, we can observe that the wrinkles at the error plots at the bottom row have noticeably smaller size than the ones of the object at the top row.

5. Conclusions

We presented a patch-based method for 3D reconstruction of textureless deformable surfaces from a single RGB image. No explicit assumptions about the shape of the observed object are made. Instead, we learn the mapping from image patches to their corresponding geometries from real-life data via deep learning. We show that, compared to previous global methods for 3D reconstruction, our method leads to more accurate 3D reconstruction, more effective generalization to unseen objects and more efficient exploitation of the available training data. We also show qualitatively the ability of our method to handle high-resolution input data. Potential directions for future work include learning how to stitch the predicted patches, predicting the optimal patch size from the observed data as well as investigating scenarios involving textured deformable objects.

Acknowledgements: This work was partially supported by the H2020-ICT-2016-1-731869 project Co4Robots. We gratefully acknowledge the NVIDIA Corporation for the donation of a Titan V GPU and Jan Bednarik for useful discussions.

References

[1] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017. 3

[2] J. T. Barron and J. Malik. Shape, illumination, and reflectance from shading. *IEEE transactions on pattern analysis and machine intelligence*, 37(8):1670–1687, 2015. 3, 6, 7

[3] J. Bednarik, P. Fua, and M. Salzmann. Learning to reconstruct texture-less deformable surfaces from a single view. In *2018 International Conference on 3D Vision (3DV)*, pages 606–615. IEEE, 2018. 1, 3, 4, 5, 6, 7, 8

[4] A. Chhatkuli, D. Pizarro, A. Bartoli, and T. Collins. A stable analytical framework for isometric shape-from-template by surface integration. *IEEE transactions on pattern analysis and machine intelligence*, 39(5):833–850, 2017. 2

[5] G. Choe, S. G. Narasimhan, and I. So Kweon. Simultaneous estimation of near ir brdf and fine-scale surface geometry. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2452–2460, 2016. 3

[6] R. Daněřek, E. Dibra, C. Öztireli, R. Ziegler, and M. Gross. Deepgarment: 3d garment shape estimation from a single image. In *Computer Graphics Forum*, volume 36, pages 269–280. Wiley Online Library, 2017. 1, 2, 3

[7] J.-D. Durou, M. Falcone, and M. Sagona. Numerical methods for shape-from-shading: A new survey with benchmarks. *Computer Vision and Image Understanding*, 109(1):22–43, 2008. 3

[8] J. Fayad, L. Agapito, and A. Del Bue. Piecewise quadratic reconstruction of non-rigid surfaces from monocular sequences. In *European conference on computer vision*, pages 297–310. Springer, 2010. 1, 3

[9] V. Golyanik, S. Shimada, K. Varanasi, and D. Stricker. Hdm-net: Monocular non-rigid 3d reconstruction with learned deformation model. In *International Conference on Virtual Reality and Augmented Reality*, pages 51–72. Springer, 2018. 1, 2

[10] C. Häne, C. Zach, B. Zeisl, and M. Pollefeys. A patch prior for dense 3d reconstruction in man-made environments. In *2012 Second International Conference on 3D Imaging, Modeling, Processing, Visualization & Transmission*, pages 563–570. IEEE, 2012. 3

[11] N. Haouchine and S. Cotin. Template-based monocular 3d recovery of elastic shapes using lagrangian multipliers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4095–4103, 2017. 2

[12] L. Hou, D. Samaras, T. M. Kurc, Y. Gao, J. E. Davis, and J. H. Saltz. Patch-based convolutional neural network for whole slide tissue image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2424–2433, 2016. 2, 8

[13] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. CVPR '17. 2

[14] M. Janner, J. Wu, T. D. Kulkarni, I. Yildirim, and J. Tenenbaum. Self-supervised intrinsic image decomposition. In *Advances in Neural Information Processing Systems*, pages 5936–5946, 2017. 3

[15] D. F. Jiménez, D. C. Pérez, D. P. Pérez, T. Collins, and A. Bartoli. Deep shape-from-template: Wide-baseline, dense and fast registration and deformable reconstruction from a single image. *arXiv preprint arXiv:1811.07791*, 2018. 2, 4

[16] D. G. Kendall. A survey of the statistical theory of shape. *Statistical Science*, pages 87–99, 1989. 5

[17] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[18] T. Koch, L. Liebel, F. Fraundorfer, and M. Körner. Evaluation of cnn-based single-image depth estimation methods. In *European Conference on Computer Vision*, pages 331–348. Springer, 2018. 4

[19] C. Kozlov, M. Slavcheva, and S. Ilic. Patch-based non-rigid 3d reconstruction from a single depth stream. In *2018 International Conference on 3D Vision (3DV)*, pages 42–51. IEEE, 2018. 3

[20] C. Li and M. Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. ECCV '16. 2

[21] J. J. Moré. The levenberg-marquardt algorithm: implementation and theory. In *Numerical analysis*, pages 105–116. Springer, 1978. 5

[22] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *2011 IEEE International Symposium on Mixed and Augmented Reality*, pages 127–136. IEEE, 2011. 5

[23] S. Parashar, D. Pizarro, A. Bartoli, and T. Collins. As-rigid-as-possible volumetric shape-from-template. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 891–899, 2015. 2

[24] A. Pumarola, A. Agudo, L. Porzi, A. Sanfeliu, V. Lepetit, and F. Moreno-Noguer. Geometry-aware network for non-rigid shape prediction from a single view. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4681–4690, 2018. 1, 2, 5

[25] Y. Quéau, J.-D. Durou, and J.-F. Aujol. Normal integration: a survey. *Journal of Mathematical Imaging and Vision*, 60(4):576–593, 2018. 5, 6

[26] K. Rematas, T. Ritschel, M. Fritz, E. Gavves, and T. Tuytelaars. Deep reflectance maps. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4508–4516, 2016. 3

[27] C. Russell, R. Yu, and L. Agapito. Video pop-up: Monocular 3d reconstruction of dynamic scenes. In *European conference on computer vision*, pages 583–598. Springer, 2014. 3

[28] M. Salzmann and P. Fua. Linear local models for monocular reconstruction of deformable surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):931–944, 2011. 3

[29] M. Salzmann, R. Urtasun, and P. Fua. Local deformation models for monocular 3d shape recovery. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008. 1, 3

[30] S. Sengupta, A. Kanazawa, C. D. Castillo, and D. W. Jacobs. Sfsnet: Learning shape, reflectance and illuminance of facesin the wild'. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6296–6305, 2018. 3

[31] Z. Shu, E. Yumer, S. Hadap, K. Sunkavalli, E. Shechtman, and D. Samaras. Neural face editing with intrinsic image disentangling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5541–5550, 2017. 3

[32] D. Tien Ngo, S. Park, A. Jorstad, A. Crivellaro, C. D. Yoo, and P. Fua. Dense image registration and deformable surface reconstruction in presence of occlusions and minimal texture. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2273–2281, 2015. 2

[33] G. Trigeorgis, P. Snape, I. Kokkinos, and S. Zafeiriou. Face normals "in-the-wild" using fully convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 340–349, 2017. 4

[34] Y. Xiong, A. Chakrabarti, R. Basri, S. J. Gortler, D. W. Jacobs, and T. Zickler. From shading to local shape. *IEEE transactions on pattern analysis and machine intelligence*, 37(1):67–79, 2015. 3

[35] D. Zoran, D. Krishnan, J. Bento, and B. Freeman. Shape and illumination from shading using the generic viewpoint assumption. In *Advances in Neural Information Processing Systems*, pages 226–234, 2014. 3