Unsupervised 3D Reconstruction Networks

Geonho Cha[†*] Minsik Lee[‡*] Songhwai Oh[†]
Electrical and Computer Engineering, ASRI, Seoul National University, Korea[†]
Division of Electrical Engineering, Hanyang University, Korea[‡]
geonho.cha@rllab.snu.ac.kr mleepaper@hanyang.ac.kr songhwai@snu.ac.kr

Abstract

In this paper, we propose 3D unsupervised reconstruction networks (3D-URN), which reconstruct the 3D structures of instances in a given object category from their 2D feature points under an orthographic camera model. 3D-URN consists of a 3D shape reconstructor and a rotation estimator, which are trained in a fully-unsupervised manner incorporating the proposed unsupervised loss functions. The role of the 3D shape reconstructor is to reconstruct the 3D shape of an instance from its 2D feature points, and the rotation estimator infers the camera pose. After training, 3D-URN can infer the 3D structure of an unseen instance in the same category, which is not possible in the conventional schemes of non-rigid structure from motion and structure from category. The experimental result shows the state-of-the-art performance, which demonstrates the effectiveness of the proposed method.

1. Introduction

Unsupervised 3D reconstruction from 2D observations is one of the key problems in computer vision, which has numerous applications such as human computer interaction and augmented reality. Structure from motion (SfM) and non-rigid structure from motion (NRSfM) have handled the unsupervised 3D reconstruction problem, of which the goals are to reconstruct 3D trajectories and motions from 2D feature trajectories of rigid and non-rigid objects, respectively. Thanks to the significant advances [16, 29], SfM is considered as a mature area in computer vision. However, NRSfM has been generally considered as an ill-conditioned problem due to the increased degrees of freedom. To alleviate this issue, some prior information on the nature of deformation has been incorporated. For example, the shape space model [4, 7, 8] and the trajectory space model [3, 15] assume that the aligned deformations of a non-rigid instance can be represented by a low-rank matrix. However, most schemes of NRSfM have focused on the reconstruction of a specific instance of an object and have assumed smooth input trajectories.

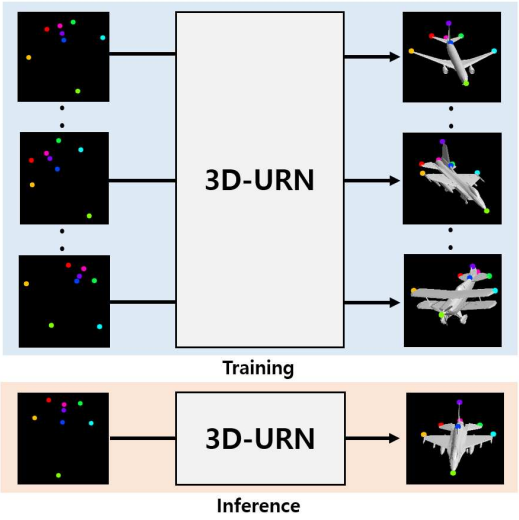

Figure 1. The proposed 3D unsupervised reconstruction networks (3D-URN) reconstructs the 3D structures of instances in an object category from their 2D feature points. After training, 3D-URN can reconstruct the 3D structure of an unseen instance in the same category as the one used in the training process.

These limitations have been alleviated in the field of structure from category (SfC) [1, 2, 12, 20], which is a problem to reconstruct the 3D structures of instances in an object category and no smoothness assumption of input feature points is needed. Furthermore, SfC can handle a general object category, unlike NRSfM that has conventionally dealt with limited object categories such as human face or body [20]. However, despite their promising results, most works of NRSfM and SfC can only produce the 3D reconstruction of a given input, i.e., the information learned from reconstructing a certain input cannot be reused, and a sufficient number of samples are needed in the reconstruction process. That is, the optimization process should be repeated with a sufficient number of samples to reconstruct

*Authors contributed equally.

the 3D structure of each instance, although the 2D feature points of instances in the same object category are similar with each other.

In this paper, we propose 3D unsupervised reconstruction networks (3D-URN), a neural-network-based framework which reconstructs 3D structures of instances in an object category from their 2D feature points under an orthographic camera model (see Figure 1). 3D-URN consists of a 3D shape reconstructor and a rotation estimator, which are based on neural networks. The role of the 3D shape reconstructor is to reconstruct the 3D shape of an instance from its 2D feature points, which is a highly non-linear procedure. To solve this problem, inspired by the shape-basis methods in NRSfM [4, 7, 8], we incorporate multiple 3D reconstructors, from which the final 3D shape is represented as a weighted sum of multiple 3D reconstructors. Here, the weights of 3D reconstructors are also estimated based on a neural-network-based estimator. On the other hand, the rotation estimator estimates a rotation matrix from the 2D feature points of an instance. Unlike the 3D shape reconstructor, the output of the rotation estimator should meet the orthogonality constraint. To handle this issue, we propose a rotation refiner that decomposes the output of the rotation estimator to make it orthogonal. The proposed rotation refiner is composed of differentiable operations, which is useful in back-propagation.

The proposed 3D-URN is trained in a fully-unsupervised manner based on the proposed loss functions, namely, the projection loss and the low-rank prior loss. After the proposed network is trained, the network can infer the 3D structure of an instance in the same object category unlike previous works of NRSfM and SfC. The proposed 3D-URN shows the state-of-the-art performance, which demonstrates the effectiveness of the scheme.

The contribution of the proposed method is summarized as:

- We propose a neural-network-based framework which solves the structure from category problem.
- We propose a novel rotation refiner, which is a differentiable layer that resolves the issues of orthogonality constraints and reflection ambiguities among the estimated rotation matrices.
- The proposed scheme shows the state-of-the-art performance on the popular benchmark data sets.

The remainder of this paper is organized as follows: we discuss related work in Section 2. The proposed 3D-URN and the proposed unsupervised loss functions are explained in Section 3. Section 4 shows the experimental results, and we conclude the paper in Section 5.

2. Related work

In this section, we introduce recent work on non-rigid structure from motion, and present some related work in structure from category.

2.1. Non-rigid structure from motion

The goal of NRSfM is to reconstruct 3D trajectories and camera poses from 2D feature trajectories of non-rigid objects. NRSfM is generally considered as an ill-conditioned problem, because of its large degrees of freedom. That is, we have to find out the camera pose and the shape of the object, which change in each frame, from an observation of 2D feature points. To resolve the ill-conditioned nature of NRSfM, some prior information has been incorporated. The low-rank assumption [3, 4, 7, 8, 15] is one of the promising prior assumptions. The local-rigidity prior [5, 6, 17, 23, 27] and the sparsity assumption [19] have also been incorporated to make the problem easier. However, most NRSfM schemes have difficulties reconstructing a set of arbitrary instances in a general object category, and so they have focused on limited object categories, such as a human face and body. Furthermore, they often assume smooth input trajectories, which limits the range of practical uses.

2.2. Structure from category

To resolve the limitations of NRSfM, some frameworks have been proposed to reconstruct multiple arbitrary instances of an object category without the assumption of smooth input trajectories [1, 2, 12, 20]. Gao et al. [12] extended two existing schemes to exploit the proposed symmetry constraints. Agudo et al. [1] proposed a formulation based on a dual low-rank shape representation, which is learned through a variant of the probabilistic linear discriminant analysis. On the one hand, Kong et al. [20] introduced the concept of SfC, of which the goal is to reconstruct the 3D shapes and camera poses of multiple instances in an object category from their 2D feature points. They formulated an augmented sparse shape-space model that can reconstruct multiple instances of a rigid object category. A thing to note here is that all of the previous works [1, 12, 20] only handled rigid object categories. To resolve this issue, Agudo et al. [2] proposed a framework for SfC which can handle both the rigid and non-rigid object categories. They formulated the model of the object deformation based on multiple unions of subspaces, which was solved with augmented Lagrange multipliers. In spite of the promising results of NRSfM and SfC, most of them can only produce the 3D reconstruction of a given input, i.e., the information learned from reconstructing a certain input cannot be reused, which hinders their utility in real world applications.

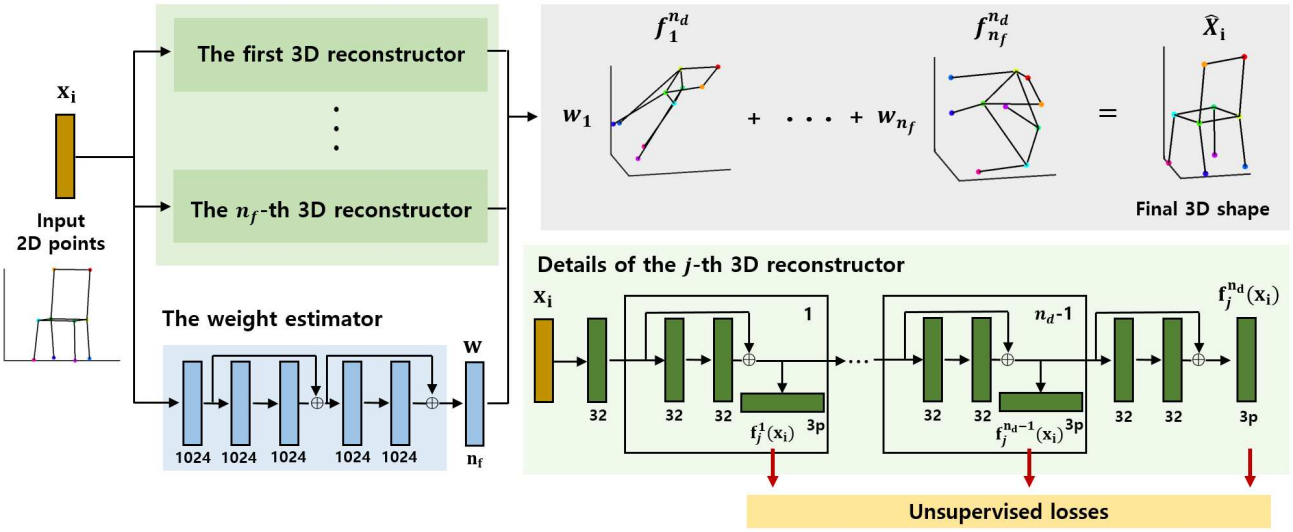

Figure 2. An overview of the proposed 3D shape reconstructor. It consists of n_f 3D reconstructors and a weight estimator. Here, each 3D reconstructor consists of n_d modules, and each module produces an intermediate reconstruction which is used in the intermediate losses to prevent the gradient vanishing problem. The reconstructions are weighted summed to estimate the final 3D shape. Here, the number written under each layer represents the channel size of the layer.

3. 3D reconstruction networks

In this section, we introduce the proposed 3D-URN, which consists of a 3D shape reconstructor and a rotation estimator. The network reconstructs the 3D shape of the i-th sample, $\hat{\mathbf{X}}_i \in \mathbb{R}^{3\times p}$, where p is the total number of points, and the camera matrix $\hat{\mathbf{R}}_i \in \mathbb{R}^{3\times 3}$ from 2D feature points of the i-th sample, $\mathbf{x}_i \in \mathbb{R}^{2\times p}$. In this paper, we assume that the 2D feature points and the reconstructed 3D shapes are translated so that their means are zero.

3.1. 3D shape reconstructor

An illustration of the proposed 3D shape reconstructor is visualized in Figure 2. The role of the 3D shape reconstructor, f, is to reconstruct the 3D shape $\hat{\mathbf{X}}_i$ given 2D feature points $\mathbf{x}_i$, which can be expressed as

$$\hat{\mathbf{X}}_i = f(\mathbf{x}_i). \tag{1}$$

In (1), the 3D shape is reconstructed based on a single reconstructor. However, 3D reconstruction from a 2D cue is a highly non-linear procedure, which might be hard to deal with based on a single reconstructor. To resolve this issue, we incorporate multiple reconstructors, and the final result is estimated as a weighted sum of reconstructions, which can be written as

$$\hat{\mathbf{X}}_i = \sum_{j=1,\cdots,n_f} \mathbf{w}_j f_j(\mathbf{x}_i), \tag{2}$$

where f_j is the j-th reconstructor, $\mathbf{w} \in \mathbb{R}^{n_f}$ is the weight of the reconstructions which is estimated from a weight estimator, $\mathbf{w}_j$ is the j-th element of $\mathbf{w}$, and n_f is the total number of reconstructors. The design of the 3D shape reconstructor has been inspired by the shape-space-based method in NRSfM [4, 7, 8] that represents the 3D shape based on a weighted sum of basis shapes. However, in their formulation, a batch of 2D feature points is always necessary to infer the basis shapes, unlike the proposed scheme which generates several basis shapes for each frame. In the next section, the details of the proposed 3D reconstructor and the weight estimator are introduced.

3.1.1 Design of the 3D reconstructor

The proposed 3D reconstructor is designed based on a neural network. The vectorized 2D feature points, $\mathrm{vec}(\mathbf{x}_i)$, where $\mathrm{vec}(\cdot)$ denotes the vectorization operator, is mapped into a feature space with a fully-connected layer, which is fed into n_d cascaded modules. Each module consists of two fully-connected layers. After each fully-connected layer, a ReLU activation function is followed. Note that the output of each module is fed into another fully-connected layer without activation function to infer intermediate result, which is denoted by $f_j^k(\mathbf{x}_i) \in \mathbb{R}^{3p}, k = 1, \cdots, n_d$. Here, the output of the last module is the final reconstruction of each 3D reconstructor, i.e., $f_j(\mathbf{x}_i) = f_j^{n_d}(\mathbf{x}_i)$.

There are two things to consider in designing the 3D reconstructor. The first one is scale ambiguity between the weights and the reconstructions in (2). Specifically, numerous combinations of $\mathbf{w}_j$ and $f_j(\mathbf{x}_i)$ with different scales can produce the same 3D shape $\hat{\mathbf{X}}_i$, which can make the solution ambiguous. To resolve this issue, we normalize the reconstructions so that they have unit Frobenius norms. The

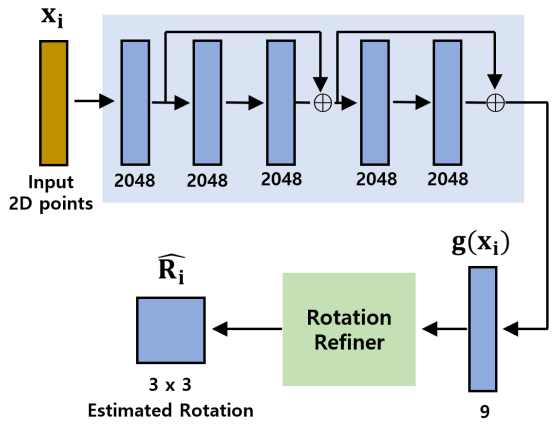

Figure 3. An overview of the proposed rotation estimator. The output of the estimation network is fed into the proposed rotation refiner, which resolves the issues of orthogonality constraints and reflection ambiguities among the estimated rotation matrices.

second issue is that the training procedure of the proposed network could suffer from gradient vanishing problem in case of large n_d. To alleviate this issue, we put the proposed losses to the output of each module, which will be introduced in Section 3.3.

3.1.2 Design of the weight estimator

An illustration of the weight estimator is visualized in Figure 2. The weight estimator infers the weights of the reconstructions from 2D feature points. It is also designed based on a neural network, which consists of six fully-connected layers. The output of each fully-connected layer is followed by a ReLU activation function except the last layer. Here, we empirically found out that incorporating the absolute value of the output from the network as the weight improves the performance. Hence, we utilize the absolute value of the output from the weight estimator as the weight, $\mathbf{w}$.

3.2. Rotation estimator

An overview of the rotation estimator is visualized in Figure 3. The role of the rotation estimator, g, is to infer rotation matrix $\hat{\mathbf{R}}_i$ given 2D feature points $\mathbf{x}_i$, which can be expressed as

$$\hat{\mathbf{R}}_i = g(\mathbf{x}_i). \tag{3}$$

The structure of the rotation estimation network is similar to the weight estimation network, with the different size of channel dimensions. However, in contrast to the design of the weight estimation network, there are two things to consider in designing the rotation estimation network. The first one is the orthogonality constraint of the rotation matrix:

$$\hat{\mathbf{R}}_i \hat{\mathbf{R}}_i^T = \mathbf{I}, \tag{4}$$

where $\mathbf{I}$ is the identity matrix. The second one is the reflection ambiguities among the estimated rotation matrices. To resolve these issues, we propose a novel rotation refining procedure, which is applied to the output of the rotation estimation network. The proposed rotation refining procedure is given as

$$\begin{aligned} \tilde{\mathbf{R}}_i &\leftarrow \mathbf{U}\mathbf{V}^T, \\ \hat{\mathbf{R}}_i &\leftarrow \mathbf{U} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & |\tilde{\mathbf{R}}_i| \end{bmatrix} \mathbf{V}^T, \end{aligned} \tag{5}$$

where $g(\mathbf{x}_i) = \mathbf{U}\mathbf{\Sigma}\mathbf{V}$ is the singular value decomposition (SVD) of $g(\mathbf{x}_i)$ and $|\cdot|$ denotes the matrix determinant. The first step here enforces the orthogonality constraint. The reflection ambiguities are also resolved by putting the estimated rotation matrices to the special orthogonal group of $|\hat{\mathbf{R}}_i| = 1$, which is achieved in the second step.

Note that many deep learning tools like TensorFlow provide an ability to back-propagate through SVD, which enables us to use the proposed rotation refiner without any problem.

3.3. Unsupervised losses

We train the proposed network in a fully-unsupervised manner incorporating the proposed losses, which are the projection loss and the low-rank loss. Before explaining the proposed loss functions, we introduce some additional notations. Let $\mathbf{S}_j^k$ be the collection of reconstructions, which are estimated from the k-th module of the j-th reconstruction network, given a batch of training samples, which is defined as

$$\mathbf{S}_j^k = \left[\mathrm{vec}(f_j^k(\mathbf{x}_1)), \quad \cdots, \quad \mathrm{vec}(f_j^k(\mathbf{x}_{n_b}))\right]^T, \tag{6}$$

where n_b is the batch size. Meanwhile, the weighted sum of the intermediate reconstructions is denoted by $\hat{\mathbf{X}}_i^k$, i.e., $\hat{\mathbf{X}}_i^k = \sum_{j=1,\cdots,n_f} \mathbf{w}_j f_j^k(\mathbf{x}_i)$.

3.3.1 Projection loss

The projection loss measures how well the projection of the rotated reconstruction coincide with the given 2D feature points. The projection loss is defined as

$$L_{\text{proj}} = \frac{1}{pn_d n_b} \sum_{i \in B} \sum_{k=1,\cdots,n_d} \|\mathbf{P}\hat{\mathbf{R}}_i \hat{\mathbf{X}}_i^k - \mathbf{x}_i\|_F^2, \tag{7}$$

where $\|\cdot\|_F$ denotes the Frobenius norm, $\mathbf{P}$ is the projection matrix, and B is a set of training batch samples. Here, we assume that 2D observations are given from the orthographic camera model, which is reasonable to model objects that are far enough from the camera compared to the depth variation. Hence, the projection matrix $\mathbf{P}$ is given as

$$\mathbf{P} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}. \tag{8}$$

3.3.2 Low-rank prior loss

The low-rank assumption has been generally incorporated in the field of NRSfM [3,4,7,8,15] to reconstruct 3D shapes in an unsupervised manner. In general, a smooth surrogate of the rank cost such as log-determinant and the nuclear norm have been adopted in the formulation [10]. We found out empirically that a nuclear-norm-based cost gives superior performance, which is given as

$$L_{\text{lr}} = \frac{1}{n_d n_f} \sum_{k=1,\cdots,n_d} \sum_{j=1,\cdots,n_f} \|\mathbf{S}_j^k (\mathbf{S}_j^k)^T\|_*, \quad (9)$$

where $\|\cdot\|_*$ denotes the nuclear norm.

3.3.3 Total loss

The total loss function of the proposed network is given as

$$L_{\text{total}} = \lambda_1 L_{\text{proj}} + \lambda_2 L_{\text{lr}} + \lambda_3 L_{\text{reg}}, \quad (10)$$

where λ_1, λ_2, and λ_3 are weighting parameters, and L_{reg} is the regularization loss for estimated weights which is defined as

$$L_{\text{reg}} = \|\mathbf{w}\|_F^2. \quad (11)$$

The loss function is defined on both every intermediate output and the last output of the reconstruction network, which facilitates the training procedure by preventing the gradient vanishing problem.

4. Experimental results

In this section, we show experimental results of 3D-URN on various data sets. For all experiments, we used the following parameter setting unless stated otherwise: $n_f = 12, n_d = 12, n_b = 32, \lambda_1 = 800, \lambda_2 = 80$, and $\lambda_3 = 5$.

4.1. Implementation details

We have empirically found that putting a constraint on parameters of each layer to clip their norm by 1 improves the performance. Specifically, we have normalized the parameter matrices of which the Frobenius norms are larger than 1 so that they have unit Frobenius norms. We have also found out that alternatingly updating the parameters of the 3D shape reconstructor and the rotation estimator shows better performance. Note that the rotation estimation performance might be crucial for training the 3D shape reconstructor. Hence, we updated the parameters of the rotation estimation network for 25 times before every update of the parameters of the 3D shape reconstructor. The parameters of both the networks were updated for 100 epochs based on Adam [18] with a learning rate of 1.0×10^{-3}. Lastly, in our framework, x and y coordinates correspond to the input coordinates. Hence, we concatenated the input 2D points with the estimated depths for the final output.

4.2. Quantitative evaluation

For the quantitative evaluation, we have applied 3D-URN on the PASCAL3D+ data set [30] and the face sequence [14]. In the PASCAL3D+ data set, ground truth 3D CAD models are labeled to the samples of the PASCAL VOC data set [9]. Following the practices in [2], we have evaluated the performance on the instances in eight object categories which have at least eight keypoints, and the performance is measured in terms of the normalized mean 3D error, e_{3D}^1, as in [7, 13], which is defined as

$$e_{\text{3D}}^1 = \frac{1}{\sigma n_s p} \sum_{i=1}^{n_s} \sum_{j=1}^{p} e_{ij}, \; \sigma = \frac{1}{3n_s} \sum_{i=1}^{n_s} (\sigma_{ix} + \sigma_{iy} + \sigma_{iz}), \quad (12)$$

where σ_{ix}, σ_{iy}, σ_{iz} are the standard deviations in x, y, and z coordinates of the ground truth 3D shape for the i-th instance, n_s is the total number of instances, and e_{ij} is the Euclidean distance error of the j-th point in the i-th instance. Note that we have calculated e_{ij} for both the original reconstructed shape and the reflected shape and picked the smaller one, due to the inherent reflection ambiguity. We have also considered full and clean 2D feature points in each object category, following the practice in [2]. We have compared the performance of 3D-URN to various methods [2, 7, 11, 13, 14, 21, 22, 24, 29, 31].

The results are summarized in Table 1. The proposed method shows the state-of-the-art performance on average, which demonstrates the effectiveness of the proposed method. Some reconstruction results are visualized in Figure 6. We have also evaluated 3D-URN on noisy input. For this evaluation, zero mean Gaussian noises are added to 2D feature points, of which the standard deviation of the noise is set to $0.01 d_{\max}$ where $d_{\max}$ is the maximum absolute value of the input feature points. The comparison results on the noisy data are summarized in Table 2. Also in this case, 3D-URN shows the state-of-the-art performance on average. In case the standard deviation of the noise has doubled, the average performance of 3D-URN was 0.189, which is comparable to the performance of [2] with small noise.

The face sequence is one of the popular benchmark data sets in the field of NRSfM, which is a motion capture data of facial feature points. We have compared the performance of 3D-URN to various methods [7,22,26] based on the error measure, e_{3D}^2, used in [21, 22], which is defined as

$$e_{\text{3D}}^2 = \frac{1}{n_s} \sum_{i=1}^{n_s} \frac{\|\mathbf{X}_i^{\text{GT}} - \mathbf{X}_i^{\text{infer}}\|_F}{\|\mathbf{X}_i^{\text{GT}}\|_F}, \quad (13)$$

where $\mathbf{X}_i^{\text{GT}}$ and $\mathbf{X}_i^{\text{infer}}$ are the ground truth 3D shape and the reconstructed 3D shape, respectively. Again, for this measure, both the original reconstructed shape and the reflected shape have been used in calculating the error. Table 3 summarizes the comparison results. Although the other NRSfM

	TK [29]	MC [24]	CSF [13]	KSTA [14]	BMM [7]	EM-PND [21]	TUS [31]	GBNR [11]	CNR [22]	MUS [2]	Ours	Ours*
Aeroplane	0.679	0.584	0.363	0.145	0.843	0.578	0.294	-	0.263	0.261	**0.121**	0.157
Bicycle	0.309	0.440	0.424	0.442	0.308	0.763	0.182	0.221	-	**0.178**	0.328	0.305
Bus	0.202	0.238	0.217	0.214	0.300	1.048	0.129	0.214	-	0.113	**0.097**	0.105
Car	0.239	0.256	0.195	0.159	0.266	0.496	0.084	0.217	0.099	**0.078**	0.104	0.097
Chair	0.356	0.447	0.398	0.399	0.357	0.687	0.211	-	-	0.210	**0.115**	0.141
Diningtable	0.386	0.512	0.406	0.372	0.422	0.670	0.265	0.351	-	0.264	**0.115**	0.107
Motorbike	0.339	0.346	0.278	0.270	0.336	0.740	0.228	0.268	-	**0.222**	0.287	0.265
Sofa	0.381	0.390	0.409	0.298	0.279	0.692	0.179	0.264	0.214	**0.167**	0.181	0.153
Average	0.361	0.402	0.336	0.287	0.388	0.709	0.196	0.256	0.192	0.186	**0.168**	0.166

Table 1. Results on the PASCAL3D+ data set based on the error measure e^1_{3D}. The performances of the other methods are quoted from [2]. Here, "-" means failure of reconstructions, and "Ours*" is the performance of 3D-URN measured on unseen instances.

	TK [29]	MC [24]	CSF [13]	KSTA [14]	BMM [7]	EM-PND [21]	TUS [31]	GBNR [11]	CNR [22]	MUS [2]	Ours	Ours*
Aeroplane	0.677	0.583	0.233	0.183	0.566	0.760	0.297	-	0.294	0.271	**0.144**	0.163
Bicycle	0.308	0.442	0.455	0.457	0.307	0.808	0.195	0.231	-	**0.188**	0.320	0.355
Bus	0.204	0.241	0.227	0.218	0.255	1.197	0.139	0.223	-	0.122	**0.103**	0.127
Car	0.241	0.259	0.169	0.164	0.161	0.624	0.100	0.222	0.122	**0.093**	0.119	0.109
Chair	0.358	0.447	0.398	0.396	0.258	0.818	0.221	-	-	0.220	**0.125**	0.151
Diningtable	0.392	0.522	0.414	0.383	0.358	0.807	0.268	0.370	-	0.267	**0.121**	0.113
Motorbike	0.342	0.348	0.295	0.290	0.299	0.748	0.237	0.277	-	**0.233**	0.295	0.240
Sofa	0.384	0.392	0.303	0.294	0.240	0.726	0.188	0.271	0.228	**0.174**	0.185	0.157
Average	0.363	0.404	0.312	0.298	0.305	0.811	0.206	0.266	0.215	0.196	**0.177**	0.177

Table 2. Results on the PASCAL3D+ data set with noise based on the error measure e^1_{3D}. The performances of the other methods are quoted from [2]. Here, "-" means failure of reconstructions, and "Ours*" is the performance of 3D-URN measured on unseen instances.

schemes show better performance, the performance of 3D-URN is comparable and is good enough for practical applications. This result shows that 3D-URN can reconstruct not only the instances in rigid object categories but also in non-rigid object categories.

On the one hand, to verify the fact that the proposed 3D-URN can reconstruct the 3D shape of an unseen instance after we train the proposed network, we have divided the instances of each object category into a training set and a test set. Specifically, we put 80% of the instances into the training set, and the remaining instances into the test set. The results on clean and noisy inputs are summarized in Table 1 and Table 2, respectively, which are represented as "Ours*". We can verify that 3D-URN is capable of reconstructing the 3D structures of unseen instances.

4.3. Qualitative evaluation

We have evaluated the proposed framework on MUCT data set [25] and the two cloths sequence [28][1] for the qualitative evaluation. MUCT is a face image data set which has labeled 2D landmark feature points. It consists of 86 RGB face images of diverse ages, races, and lighting conditions, and each image has 68 2D feature points. On the one hand, the two cloths sequence is one of the popular sequences in the field of NRSfM, which is a sequence of two independently moving cloths. It consists of 163 frames, and each frame has 525 points. In this case, we set $n_f = 24$ to handle the large number of 2D feature points. Since no ground truth 3D points are provided for both cases, we only report the qualitative results. An illustration of the reconstruction results on MUCT data set is visualized in Figure 4. We can verify that the proposed scheme has successfully reconstructed the 3D shapes of faces. Some reconstruction results of the two cloths sequence are visualized in Figure 5. A notable thing is that 3D-URN has successfully reconstructed multiple parts of an instance at once without a separate segmentation process. Again from the results of the MUCT data set and the two cloths sequence, we can verify that 3D-URN can reconstruct instances in non-rigid object categories.

[1]The reconstruction video is provided in the supplementary material.

4.4. Additional experiments

We have performed ablation experiments on the PASCAL3D+ data set. The performance was evaluated by removing some components of the proposed scheme. The results are summarized in Table 4. Removing the low-rank prior loss increased the error by 0.007. The error was increased by 0.006 when we removed the weight regularization loss, and by 0.100 when the intermediate loss was not incorporated.

In addition, we have evaluated the performance of 3D-URN on the PASCAL3D+ data set with various numbers of 3D reconstructors, of which the results are summarized in Table 5. We can see that the performance gets improved as the number of 3D reconstructors increases, which shows the validity of incorporating multiple 3D reconstructors. Lastly, we have performed experiments on the PASCAL3D+ data set with other low-rank losses. The use of the determinant cost increases the error by 0.009, and using the log-determinant cost causes numerical instability, resulting in a training failure.

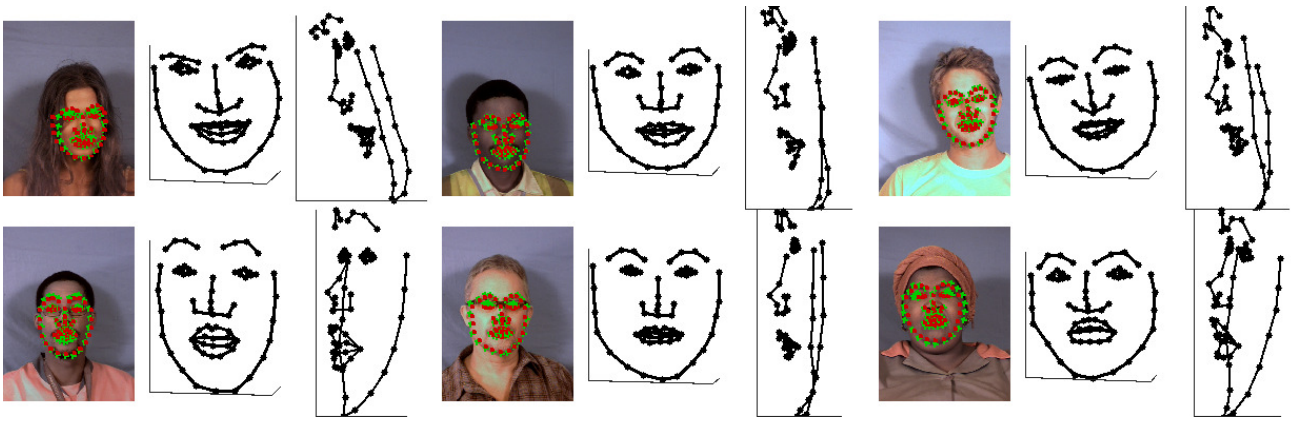

Figure 4. Some example reconstructions of the MUCT data set. Left: It shows the input 2D feature points and the projection of the reconstructed 3D shape. Here, green circle means the input 2D feature point, and the red "+" means the estimated 2D feature point. Middle, Right: They visualize the reconstructed 3D shape in the camera view and the side view.

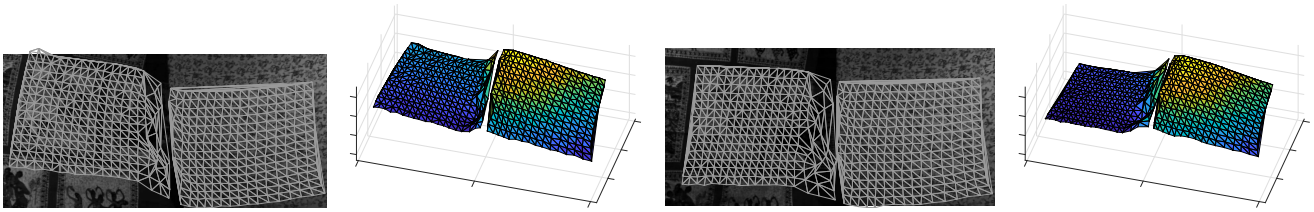

Figure 5. Some example reconstructions of the two cloths sequence. In each pair, the left figure represents the input 2D feature points, and the right figure shows the reconstruction result.

	MP [26]	BMM [7]	CNR [22]	Ours
Error	0.032	0.023	0.025	0.044

Table 3. Results on the face sequence based on the error measure e^2_{3D}.

Variant	Ours	w/o L_{lr}	w/o L_{reg}	w/o L_{int}
Error	0.168	0.175	0.174	0.268
$\triangle$	-	0.007	0.006	0.100

Table 4. Ablation experiments on different components in our framework.

n_f	2	3	5	9	10	12
Error	0.257	0.211	0.211	0.211	0.176	0.168

Table 5. Average errors for various numbers of 3D reconstructors on the PASCAL3D+ data set.

5. Conclusion

In this paper, we have proposed 3D-URN, a 3D structure reconstruction network of which the goal is to reconstruct 3D structures of instances in an object category given their 2D feature points. The proposed network consists of a 3D shape reconstructor and a rotation estimator, which are trained in a fully-unsupervised manner based on the proposed loss functions. The 3D shape reconstructor reconstructs the 3D shape of an instance from 2D feature points, which is a highly non-linear procedure. To alleviate the difficulty, we incorporated multiple 3D reconstructions, which are weighted summed to estimate the final 3D shape. The rotation estimator estimates the camera pose of the given instance. To handle the orthogonality constraint and the reflection ambiguities in the rotations, we proposed a rotation refiner, which consists of differentiable operations. The proposed method shows the state-of-the-art performance on the popular PASCAL3D+ data set, demonstrating the effectiveness of the proposed scheme. Meanwhile, reconstructing multiple object categories simultaneously with a single network and extending the proposed scheme to handle missing data and the temporal smoothness assumption are key problems that can broaden the practical applications, which are left as future work.

Acknowledgment

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2019-0-01190, [SW Star Lab] Robot Learning: Efficient, Safe, and Socially-Acceptable Machine Learning, and No. 2019-0-01371, Development of Brain-Inspired AI with Human-Like Intelligence).

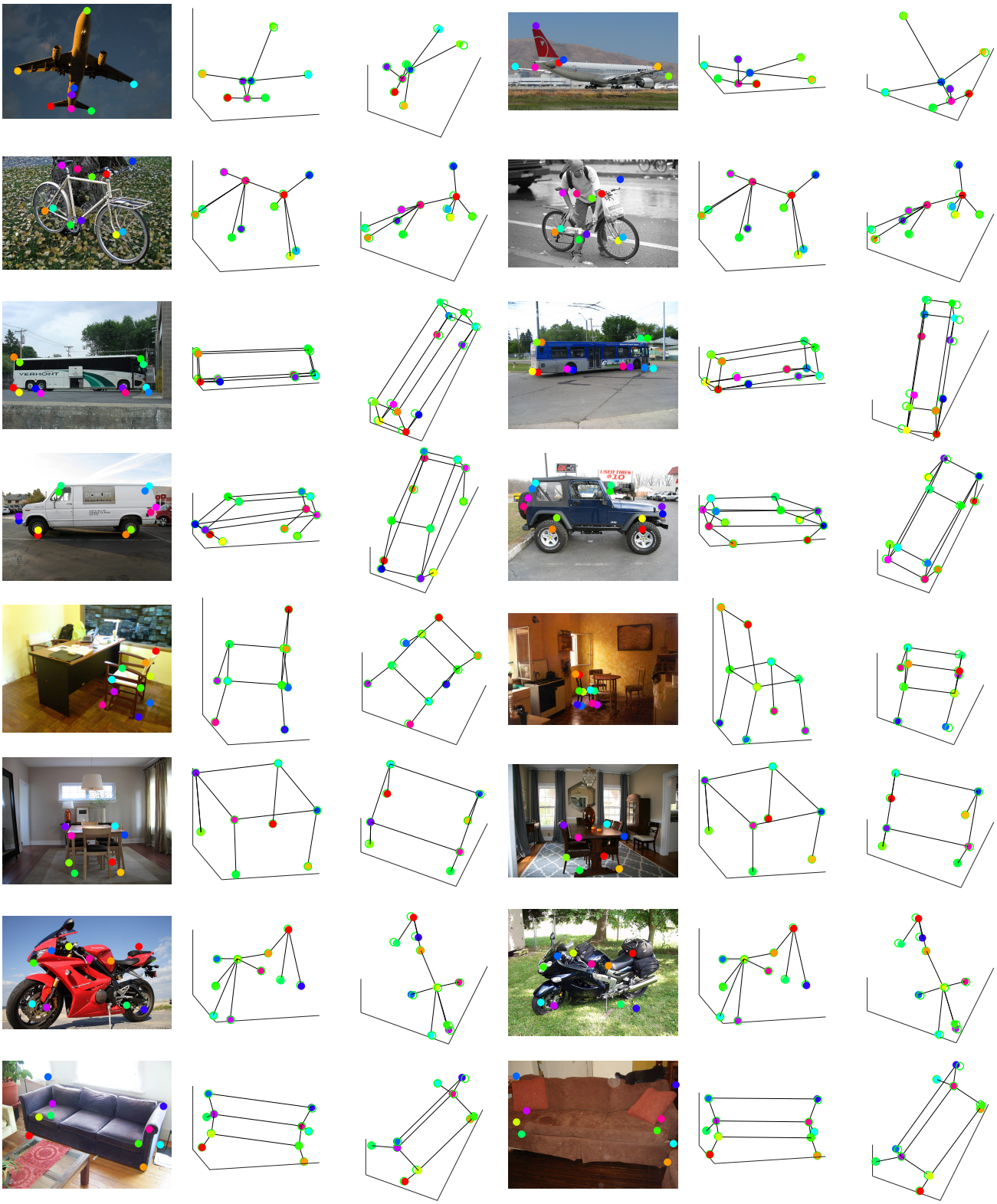

Figure 6. Some example reconstructions of the PASCAL3D+ data set. Left: Given 2D feature points, Middle, Right: 3D reconstruction results in two different views. Here, the empty circles represent the ground-truth 3D structures, and the filled circles represent the reconstructed 3D shapes.

References

[1] Antonio Agudo and Francesc Moreno-Noguer. Recovering Pose and 3D Deformable Shape from Multi-instance Image Ensembles. In *Proc. of the Asian Conference on Computer Vision*, pages 291–307, 2016.

[2] Antonio Agudo, Melcior Pijoan, and Francesc Moreno-Noguer. Image Collection Pop-up: 3D Reconstruction and Clustering of Rigid and Non-Rigid Categories. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2018.

[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.

[4] Christoph Bregler, Aaron Hertzmann, and Henning Biermann. Recovering Non-Rigid 3D Shape from Image Streams. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2000.

[5] Geonho Cha, Minsik Lee, Jungchan Cho, and Songhwai Oh. Non-rigid surface recovery with a robust local-rigidity prior. *Pattern Recognition Letters*, 110:51–57, July 2018.

[6] Ajad Chhatkuli, Daniel Pizarro, Toby Collins, and Adrien Bartoli. Inextensible Non-Rigid Shape-from-Motion by Second-Order Cone Programming. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2016.

[7] Yuchao Dai, Hongdong Li, and Mingyi He. A Simple Prior-free Method for Non-Rigid Structure-from-Motion Factorization. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2012.

[8] Alessio Del Bue. A Factorization Approach to Structure from Motion with Shape Priors. In *Proc. Computer Vision and Pattern Recognition*, June 2008.

[9] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The PASCAL Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

[10] Maryam Fazel, Haitham Hindi, and Stephen P Boyd. Log-det heuristic for matrix rank minimization with applications to Hankel and Euclidean distance matrices. In *Proc. American Control Conference*, July 2003.

[11] Katerina Fragkiadaki, Marta Salas, Pablo Arbeláez, and Jitendra Malik. Grouping-Based Low-Rank Trajectory Completion and 3D Reconstruction. In *Advances in Neural Information Processing Systems*, pages 55–63, 2014.

[12] Yuan Gao and Alan L Yuille. Symmetric Non-Rigid Structure from Motion for Category-Specific Object Structure Estimation. In *Proc. of the European Conference on Computer Vision*, pages 408–424, 2016.

[13] Paulo F. U. Gotardo and Aleix M. Martinez. Computing Smooth Time-Trajectories for Camera and Deformable Shape in Structure from Motion with Occlusion. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 33(10):2051–2065, October 2011.

[14] Paulo F. U. Gotardo and Aleix M. Martinez. Kernel Non-Rigid Structure from Motion. In *Proc. IEEE Int'l Conf. Computer Vision*, November 2011.

[15] Paulo F. U. Gotardo and Aleix M. Martinez. Non-Rigid Structure from Motion with Complementary Rank-3 Spaces. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2011.

[16] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2 edition, 2004.

[17] Pan Ji, Hongdong Li, Yuchao Dai, and Ian Reid. Maximizing Rigidity Revisited: a Convex Programming Approach for Generic 3D Shape Reconstruction from Multiple Perspective Views. In *Proc. IEEE Int'l Conf. Computer Vision*, October 2017.

[18] Diederik P Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[19] Chen Kong and Simon Lucey. Prior-less Compressible Structure from Motion. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2016.

[20] 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*, October 2016.

[21] Minsik Lee, Jungchan Cho, Chong-Ho Choi, and Songhwai Oh. Procrustean Normal Distribution for Non-Rigid Structure from Motion. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2013.

[22] Minsik Lee, Jungchan Cho, and Songhwai Oh. Consensus of Non-Rigid Reconstructions. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2016.

[23] Xiu Li, Hongdong Li, Hanbyul Joo, Yebin Liu, and Yaser Sheikh. Structure from Recurrent Motion: From Rigidity to Recurrency. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2018.

[24] Manuel Marques and Joao Costeira. Optimal shape from motion estimation with missing and degenerate data. In *IEEE Workshop on Motion and video Computing*, January 2008.

[25] Stephen Milborrow, John Morkel, and Fred Nicolls. The MUCT Landmarked Face Database. *Pattern Recognition Association of South Africa*, 2010.

[26] Marco Paladini, Alessio Del Bue, Marko Stošić, Marija Dodig, Jo ao Xavier, and Lourdes Agapito. Factorization for Non-Rigid and Articulated Structure using Metric Projections. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2009.

[27] Daniel Pizarro Shaifali Parashar and Adrien Bartoli. Isometric Non-Rigid Shape-from-Motion in Linear Time. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2016.

[28] Jonathan Taylor, Allan D. Jepson, and Kiriakos N. Kutulakos. Non-rigid Structure from Locally-rigid Motion. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2010.

[29] Carlo Tomasi and Takeo Kanade. Shape and Motion from Image Streams under Orthography: a Factorization Method. *Int'l J. Computer Vision*, 9(2):137–154, November 1992.

[30] Yu Xiang, Roozbeh Mottaghi, and Silvio Savarese. Beyond PASCAL: A Benchmark for 3D Object Detection in

the Wild. In *Porc. IEEE Winter Conference on Applications of Computer Vision*, March 2014.

[31] Yingying Zhu, Dong Huang, Fernando De La Torre, and Simon Lucey. Complex Non-rigid Motion 3D Reconstruction by Union of Subspaces. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2014.