DR-KFS: A Differentiable Visual Similarity Metric for 3D Shape Reconstruction

Jiongchao Jin[1,2], Akshay Gadi Patil[2], Zhang Xiong[1], and Hao Zhang[2(✉)]

[1] Beihang University, Beijing, China
{jinjiongchao,xiongz}@buaa.edu.cn
[2] Simon Fraser University, Burnaby, Canada
{akshay_gadi_patil,haoz}@sfu.ca

Abstract. We introduce a *differential visual similarity* metric to train deep neural networks for 3D reconstruction, aimed at improving reconstruction quality. The metric compares two 3D shapes by measuring distances between multi-view images *differentiably* rendered from the shapes. Importantly, the image-space distance is also differentiable and measures visual *similarity*, rather than pixel-wise distortion. Specifically, the similarity is defined by mean-squared errors over HardNet features computed from probabilistic keypoint maps of the compared images. Our differential visual shape similarity metric can be easily plugged into various 3D reconstruction networks, replacing their distortion-based losses, such as Chamfer or Earth Mover distances, so as to optimize the network weights to produce reconstructions with better structural fidelity and visual quality. We demonstrate this both objectively, using well-known shape metrics for retrieval and classification tasks that are independent from our new metric, and subjectively through a perceptual study.

Keywords: 3D Reconstruction · Visual similarity metric · Differentiablity

1 Introduction

Reconstructing 3D structures from 2D images is one of the most fundamental problems in computer vision. The problem is clearly ill-posed, hence most reconstruction algorithms are expected to incur errors against the ground truth. Commonly used error metrics are defined by *shape distortions*, such as Chamfer Distance (CD), Earth Mover Distance (EMD), Mean Square Error (MSE), and Intersection over Union (IoU). Most deep neural networks developed for 3D reconstruction [9–11,16,26,29,33] are trained by loss functions defined by one of these shape distortion metrics. To serve as a network loss to enable back propagation, the adopted metric needs to be *differentiable*. All of CD, EMD, and MSE
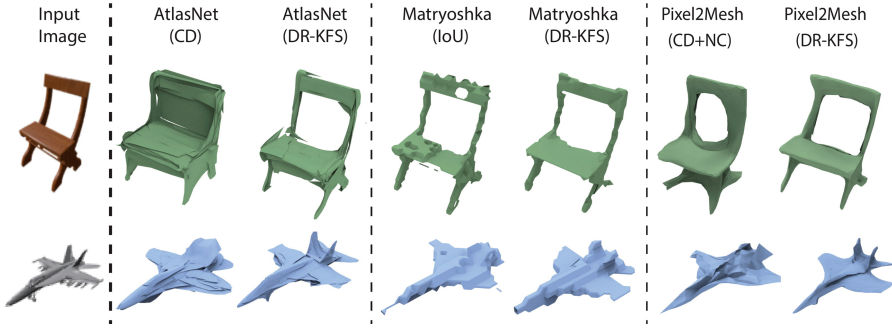
Fig. 1. Single-view 3D reconstruction by AtlasNet [10], Matryoshka Network [23], and Pixel2Mesh [29], trained with different losses. Left of each pair: trained by shape distortion in CD, IoU, and a combination of CD and normal consistency (NC). Right of each pair: using DR-KFS, our *differentiable visual similarity* metric, as train loss leads to results of higher visual fidelity in terms of shape structures and surface quality.

are differentiable. While the original IoU is not, there have been recent attempts to make it differentiable [16,35] (Fig. 1).

What is common about these well-adopted distortion metrics is that they are all designed to measure geometric distances between *aligned* 3D shapes, in the *object space*—they do not account for how the shapes are viewed by human observers, i.e., they are not "visual". Many recent works [7,9,19,22,23,27] have shown relative insensitivity of these object-space distortion metrics to *structural errors* such as missing parts and topological noise, and visual artifacts such as self intersections and poor surface quality. For example, thickening and elongating/shortening all four legs of a chair may result in a larger CD than removing one of the legs entirely, yet the latter alteration, a structural change, is more visually apparent; see Fig. 2. As a result, while some reconstructed 3D shapes do exhibit better visual quality, ratings based on distortion measures such as CD, EMD, MSE, or IoU may not reflect that superiority. This is not entirely surprising—a recent work by Blau and Michaeli [3] even suggests a trade-off between perceptual and distortion measures, albeit for image restoration.

Our key observation is that in contrast to shape distortion, measures of *visual similarity* are less sensitive to misalignment and slight shape distortion, but more sensitive to structural errors and visual quality. In computer graphics, the best known visual similarity metric is the light field descriptor (LFD) [6]. LFDs are computed for silhouette images of 3D shapes rendered from multiple viewpoints sampled around the shapes. Both a contour-based (Fourier descriptor) and a region-based (Zernike moment) image-space descriptor are employed, where rotational alignment is resolved via a discrete exhaustive search. However, due to the discrete rasterization during rendering and the use of truncated Fourier descriptors, LFD is non-differentiable. Some works in computer vision have also utilized multi-view, *projective* losses to train reconstruction networks, e.g., Perspective Transformer Net (PTN) [33], Neural Mesh Render [13], and
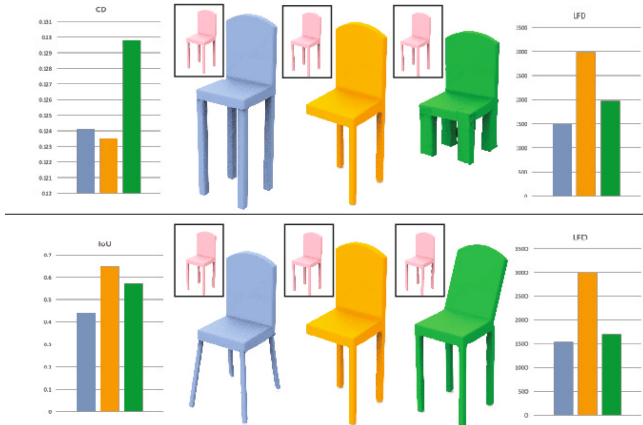
Fig. 2. A *trade-off* between shape distortion measures, e.g., CD and IoU, and a visual similarity metric such as LFD. Top: changing the shapes of the legs of a chair, e.g., by thickening and elongating (blue) or shortening (green), leads to larger shape distortions (reflected by larger CDs) compared to removing one leg (orange). The leg removal appears to make more of a visual impact due to a *structure* alteration and it is captured by a larger LFD (orange bars). Note that the bars show CDs/LFDs between the changed chair and the original, shown in pink in the boxes. Bottom: bending the legs (blue) or the back (green) leads to larger shape distortion, as reflected by *smaller* IoU, compared to leg removal. Again, the latter is captured by a larger LFD. (Color figure online)

SoftRas [16]. However, the projective- or image-space distances employed by these methods are still based on pixel-wise distortion, rather than similarity.

In this paper, we advocate the use of *differentiable visual shape similarity* metrics to train deep neural networks (DNNs) for 3D reconstruction. We introduce such a metric which compares two 3D shapes by measuring visual, image-space similarity between multi-view images rendered from the shapes, similar to LFD. However, one key difference is that the rendering process is differentiable, by employing a simplified *soft rasterization* [16]. In addition, we develop a differentiable similarity distance in the image space based on MSE defined over *probabilistic keypoint maps* of the compared images, rather than on RGB values. Furthermore, the MSE is defined over HardNet [20] features, rather than on the original keypoint maps. This choice is motivated in part by the finding in [3] that the perception-distortion trade-off appears less severe for distance between VGG features. Putting all these together, we arrive at a differentiable visual similarity metric, which we call DR-KFS to capture the use of Differentiable Rendering, Keypoint maps, and Feature-space image Similarity distances. DR-KFS better matches visual evaluation by humans, compared to object-space metrics.

To the best of our knowledge, existing 3D reconstruction networks all employ distortion-based reconstruction losses, either in the object or image spaces see Tables 1 and 2 for a summary. Our differential visual similarity metric can be

Table 1. *Object-space training losses and evaluation metrics for state-of-the-art 3D reconstruction networks.* CD and EMD are widely adopted in point-based networks, while MSE, cross-entropy, and differentially implemented IoU are more popular in voxel-based reconstruction. Implicit decoders all use per-point reconstruction loss.

Method	Representation	Training loss	Evaluation metrics
AtlasNet [10]	Points + mesh	CD	CD
PointOutNet [9]	Points	CD, EMD	CD, EMD
OGN [26]	Octree	cross-entropy/MSE	CD, EMD
HSP [11]	Voxel	cross-entropy/MSE	CD, IoU
Pixel2Mesh [29]	Points + mesh	CD, surface normal	CD, EMD
Matryoshka Net [23]	Voxel shape layer	IoU, cos-similarity	IoU
IM-Net [7]	Implicit field	Per-point status	CD, MSE, IoU, LFD
DISN [32]	Implicit field	Per-point status	CD, F-score, IoU

Table 2. *Multi-view training losses and evaluation metrics for state-of-the-art 3D reconstruction networks.* All the image-space errors are based on pixel-wise distortion such as MSE, IoU, or L1 Norm. All the evaluations are based on IoU.

Method	2D observation	Training loss	Evaluation
PTN [33]	Silhouettes	Pixel-wised MSE	IoU
NMR [13]	Silhouettes	Pixel-wised MSE	IoU
Soltani et al. [1]	Depth Maps and Silhouettes	L1 norm	IoU
SoftRas [16]	RGBA images	IoU	IoU

easily plugged into these networks, replacing the distortion losses so as to optimize the network weights to produce reconstruction results with better structural fidelity and visual quality. We demonstrate this both objectively, using well-known visual shape metrics for retrieval and classification tasks that are *independent* from DR-KFS, and subjectively through a user study.

Specifically, the 3D reconstruction networks tested for adaptation to DR-KFS training include OGN, AtlasNet [10], and Matryoshka Networks [23], which were picked as representative networks by Tatarchenko et al. [27] in their recent systematic study of single-view 3D reconstruction. We also consider Pixel2Mesh [29], NMR [13], SoftRas [16] and 3D-R2N2 [8]. The visual shape metrics we employ for evaluation include Shape Google [4], Multiview CNN or MVCNN [25], normal consistency or NC [19], F-score [23,27], as well as LFD. It is worth noting that the last three metrics have all been adopted by recent works on 3D reconstruction [7,19,23,27] to complement the distortion metrics.

2   Related Work

DNNs for 3D Reconstruction. One of the earlier DNNs for 3D reconstruction is 3D-R2N2 [8], which proposes the use of Recurrent Neural Networks (RNNs) to

reconstruct a voxelized shape from a single-view image. Another notable work, PointOutNet [9], produces multiple reconstruction candidates for a single-view input image and demonstrates an improvement in the reconstruction quality over 3D-R2N2. OGN [26] generates volumetric 3D outputs in a compute-and-memory efficient manner by using an octree representation. AtlasNet [10] uses a collection of parametric surface elements to represent a 3D shape and to naturally infer a surface representation of the shape. Pixel2Mesh [29] produces 3D triangle meshes from a single image using Graph Convolution Networks (GCN) [14]. Matryoshka Network [23] introduces a novel and efficient 2D encoding scheme for 3D geometry, posing 3D reconstruction as a 2D prediction problem, while also speeding up the process.

Recent works on generative shape modeling using implicit functions have achieved state-of-the-art visual quality. IM-NET [7] and OccNet [19] represent 3D surfaces implicitly in the form of continuous binary decision functions. DeepSDF [12] learns a continuous Signed Distance Function (SDF) representation for a 3D shape. DISN [32] combines implicit SDF descriptors with the local input image features to generate 3D shape surfaces. Despite the good visual quality of the implicit-based generated surfaces, they often obtain lower scores on *shape distortion* metrics, such as CD.

Object-Space Metrics for 3D Reconstruction. Table 3 lists state-of-the-art 3D reconstruction DNNs trained using object-space losses, along with the shape metrics they employed. In principle, a network that is trained on a particular loss, e.g., CD, should be expected to perform better on that metric during testing, than methods that were not trained with the metric. Most methods that we are aware of train their networks using object-space distortion metrics. CD and EMD, and to a lesser degree, IoU, are the dominant measures applied for evaluating reconstruction quality. It is interesting to observe that some of the most recent methods, including IM-Net [7], OccNet [19], and Matryoshka Network [23], have opted to evaluate their methods using alternative measures such as LFD, Normal Consistency (NC), and F-score. Not surprisingly, each of these works pointed out the shortcomings of object-space distortion metrics in capturing visual similarity, which motivated their use of other alternatives. In terms of visual quality of the 3D reconstruction, the current state-of-the-art results are obtained by methods based on learning implicit fields [7,12,19,32].

Visual Shape Similarity Measures. For shape retrieval and classification, many visual shape similarity measures have been proposed. LFD [6] performs visual shape similarity by extracting local features from one-hundred orthogonal projections of each 3D model. ShapeGoogle [4] constructs compact and informative shape descriptor using a *bag-of-visual- features*. MVCNN [25] combines information from multiple views of a 3D shape into a single and compact shape descriptor with the help of CNNs. F-score [27] performs visual shape similarity explicitly by calculating the distance between object surfaces using a harmonic mean of precision and recall scores. And finally, normal consistency (NC) [19,29] measures how well any 3D reconstruction can capture higher order information

by calculating a mean absolute dot product of face-normals of the given 3D models, represented as meshes. However, LFD, ShapeGoogle, F-score are not differentiable, and NC can only capture higher-order information.

Image-Space Similarity. Most common quantitative measures of image similarities include pixel-wise MSE and IoU. However, such measures are not designed to compare visual similarity between objects captured in the images, e.g., they are sensitive to misalignment. Hand-crafted image descriptors such as SIFT [17], SURF [2], and ORB [24] can be used to perform image retrieval to visually correspond closest image matches, but they are discrete and non-differentiable. On the other hand, the use of powerful image descriptors obtained from CNNs has been shown to make the image retrieval pipeline differentiable [18,21,25,34], allowing an end-to-end neural network approach for measuring image similarity. Generic image similarity has to account for rotation, translation, and occlusion of identical entities in the two images. Our aim is to develop a differentiable and visual image-similarity metric for rendered images of aligned 3D shapes.

Multi-view/Projective Approaches to 3D Reconstruction. Prior works rely on either a differentiable renderer or multi-view projections of 3D shapes and perform supervised learning on the obtained images rather than in the 3D object space, e.g., PTN [33]. Table 2 lists notable recent works along these lines and the losses and metrics they employed for training and evaluation. Specifically, NMR [13], which outperformed PTN, reconstructs a 3D mesh from a single image with silhouette-image supervision. Other works such as MarrNet [31] and Soltani et al. [1] use 2.5D sketch images and rendered silhouette images, respectively, for learning single-view 3D reconstruction. Most recently, SoftRas [16] reconstructs a 3D shape using shaded images, i.e., view-renderings of the input RGB image with a light source. Importantly, SoftRas converts the discrete raster operation to a probabilistic soft raster one, directly solving the indifferentiable raster operation of the traditional renderer present in [13,33]. However, in all these works, the image-space errors are still measured by distortion losses such as pixel-wise MSE or IoU, which do not capture shape similarity. In contrast, our metric, DR-KFS, is designed to capture visual similarity of the captured shapes. The renderer used in DR-KFS is a simplified version of SoftRas [16], which is more efficient than CNN-based methods for object rendering.

3 Differential Visual Shape Similarity Metric

Visual shape similarity metrics, such as LFD [6], rely on projections or renderings of multi-view images of 3D models from a set of view angles. We follow this theme of render-and-match-images to compare the visual similarity between two 3D shapes while make the framework *differentiable* so that it can be easily plugged into any 3D reconstruction DNN; see Fig. 4 for an overall pipeline.

Given two 3D models, one reconstructed using an existing approach such as AtlasNet [10], Matryoshka Network [23], Pixel2Mesh [29], or OGN [26], and the
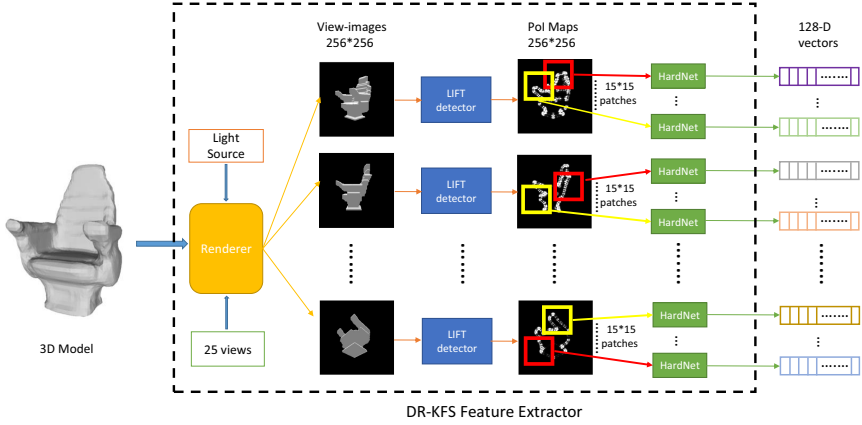
Fig. 3. Overall framework for obtaining DR-KFS descriptors using continuous patch features. The first step is the differentiable renderer, a *Soft Rasterizer*, which renders multi-view images. This renderer, coupled with the process of continuous patch feature extraction over the point-of-interest (PoI) maps, makes our framework differentiable.

other being the corresponding ground truth (GT) model, we first align them and render the models from twenty-five viewing angles, using the *Soft Rasterizer* [16] renderer, which is differentiable. For each view-image, a Point of Interest (PoI) map is obtained using a keypoint detection network. Visual similarity of the 3D models is determined using DR-KFS feature matching (see Fig. 4) obtained by extracting local features from PoI maps.

3.1   Differentiable Renderer

To measure 3D shape similarity using LFD, projected images from many angles are required. Due to hardware constraints in training our DNNs, we only use 25 viewpoints, sampled on a semi-sphere as an approximation to the ideal case. To achieve differentiability in the rendering process, we replace rasterization and z-buffering operations used in conventional rendering pipelines with soft rasterization and probabilistic map aggregation [16]. No color information is used in DR-KFS renderer as many reconstruction networks are incompatible with texture projections on the reconstructed model surfaces. A fixed position light source is aptly placed to provide information about shape surface quality when rendering the view-images. We make use of the rendered view-images to determine the similarity of given 3D models, as explained below.

Our visual similarity metric is based on matching rendered view-images by extracting discriminative local features over Point of Interest (PoI) maps, using Convolutional Neural Networks (CNNs).

PoI Map. A PoI map is a probabilistic map that gives a score for every pixel being a keypoint. It is generated by a CNN-based keypoint detection network
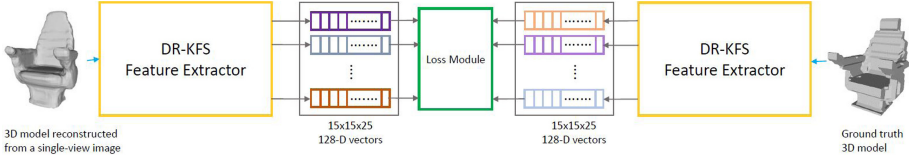
Fig. 4. An end-to-end pipeline of our visual, differentiable shape-similarity metric framework for single view 3D reconstruction. Feature descriptors from Fig. 3 are used to calculate the overall loss, which is backpropagated to the 3D reconstruction network (which reconstructs a 3D model, shown on the left) via the associated DR-KFS module, improving the reconstruction quality.

corresponding to every view-image. To this end, we borrow the LIFT detector introduced in [34], which uses piece-wise linear activation functions [28] in convolution layers to get a PoI map for an input image. It is formulated as:

$$PoI_{map} = f_{Net}(I) = \sum_{n}^{N} \delta_n \max_{m}^{M}(W_{mn} \circledast I + b_{mn}) \tag{1}$$

where $f_{Net}(I)$ is a non-linear function of the rendered view-image I, using a neural network Net, which is nothing but a CNN-based keypoint detector. N, M are hyperparameters controlling the complexity of the piece-wise linear activation function. δ is $+1$ if n is odd, and -1 otherwise. The parameters of the network Net to be learned are the convolution filter weights W_{mn} and biases b_{mn}, and $\circledast$ denotes the convolution operation. A detailed description of LIFT based keypoint detection can be found in [34].

Continuous Patch Features. After generating PoI maps for each of the rendered images, local patch features are extracted from them using a sliding window of size 32×32 with a stride of 16, resulting in 225 (15×15) patches per image I. These patches span the entire image and there exists an overlap between adjacent patches which provides continuity over the feature space.

Note that in existing image retrieval works, such as in [28,34], detected keypoints are projected back onto the original image and image features are extracted locally around these keypoints. Image retrieval is performed by matching these keypoint-based local features. This is a *discrete* process over the image space. As a result, this approach of matching images based on discrete keypoints, if employed, introduces non-differentiability into the latter parts. Unlike such methods, we avail continuous local patch features from the PoI maps by employing a sliding window technique spanning the entire map. Our framework is inspired by the recent work of [3] which shows that local patch features help alleviate the problem of "human perception *vs* image distortion" trade-off.

A sliding window on a PoI map, followed by the HardNet[20] feature extractor outputs a 128-D feature descriptor per patch (see Fig. 3). The HardNet [20] module we use is pretrained on UBC PhotoTour [30]. We coin these features as the DR-KFS feature descriptors. The continuous patch features, coupled with

the early-stage render (*Soft Rasterizer*) allow our pipeline to be entirely differentiable, demonstrated further by our end-to-end learning framework in Fig. 4.

Localized Feature Matching. After extracting the DR-KFS feature descriptors for the two 3D models, we perform localized two-way feature matching. For every patch P_{recon} centered at (x, y) in the PoI maps corresponding to the reconstructed model, we consider a set of neighboring patches in the GT PoI maps, centered at $(x \pm \sigma, y \pm \sigma)$, and find a patch that yields the minimum MSE over the patch features. We then perform a weighted addition of all such minimal loss values to get a similarity score as given in Eq. 2.

$$Sim_{recon} = \sum w_i |P_{recon_i} - P_{gt_k}| \tag{2}$$

where $w_i = average(P_{recon_i})$. P_{gt_k} is the kth patch in the PoI map corresponding to the GT model which has the least patch-feature MSE for P_{recon_i}. It should be noted that the search for the best match from the PoI patches of the ground truth model for a PoI patch of the reconstructed model is limited to PoI patches from the same viewpoint. The reconstructed features would not all converge to a single GT feature since our search is localized: search for a matching patch is done within a local neighborhood. However, the matching is many-to-one in general, which is by design. For example, due to scale discrepancies between the reconstructed and the GT models, e.g., a chair with long back vs. one with a short back, several patches of one model (e.g., on the long back) may be best matched with a single sampled patch on the other model (e.g., the short back).

We essentially repeat the above procedure for the PoI maps corresponding to the GT model and get a similarity score Sim_{GT}. Our final training loss is the sum of Sim_{recon} and Sim_{GT}, which is backpropagated through the DR-KFS feature descriptor module associated with the reconstructed 3D model (Fig. 4).

4   Results and Evaluation

We assess the impact of our differentiable visual similarity metric, DR-KFS, both qualitatively and quantitatively, on state-of-the-art single-view 3D reconstruction networks. A gallery of qualitative comparison on results generated when such networks are trained using DR-KFS is shown in Fig. 7.

4.1   Quantitative Evaluation

We consider existing single-view 3D reconstruction networks, including 3D-R2N2 [8], OGN [26], AtlasNet [10], Matryoshka Network [23], Pixel2Mesh [29], NMR [13] and SoftRas [16] to quantify both (object-space) shape distortion (CD and IoU) and the visual quality of the reconstructed models. The networks are trained either using their original losses or our metric DR-KFS. For visual quality measures, we choose representative tools including Shape Google, Normal Consistency, F-score, LFD, and MVCNN.

Table 3. Quantitative comparison for single-view 3D reconstruction networks when trained using their original distortion-based losses vs. DR-KFS, our differentiable visual similarity metric. Tested DNNs include AtlasNet25 [10] (Atlas25), Matryoshka Net [23] (MN), Pixel2Mesh [29] (P2M), Octree Generating Network [26] (OGN), 3D-R2N2 [8], and shape generation using NMR [13] and SoftRas [16]. Evaluation is done on both object-space (CD and IoU) and visual similarity metrics: SG = Shape Google, FS = F score, NC = Normal Consistency, LFD = Light Field Descriptor, MVCNN = multi-view CNN. Numbers reported represent average performance over four object categories.

Networks	Training strategies	Evaluation metrics							
		CD	IoU	SG	FS	NC	LFD	MVCNN	DR-KFS
Atlas25 [10]	Original	**6.53**	53.38	3.17	63.17	**0.792**	4338	0.532	0.288
	DR-KFS	7.11	**54.41**	**2.26**	**65.48**	0.790	**3796**	**0.522**	**0.213**
MN [23]	Original	**2.86**	65.33	4.32	**74.22**	0.825	3866	**0.637**	0.300
	DR-KFS	2.91	**66.08**	**2.08**	65.30	**0.829**	**3675**	0.660	**0.265**
P2M [29]	Original	**6.61**	54.08	6.08	58.15	**0.761**	4508	**0.648**	0.463
	DR-KFS	6.98	**54.32**	**3.84**	**60.23**	0.754	**4212**	0.662	**0.332**
OGN [26]	Original	6.13	**57.01**	9.94	**56.36**	0.695	4436	0.752	0.522
	DR-KFS	**6.08**	56.83	**8.73**	55.09	**0.702**	**4237**	**0.661**	**0.461**
3D-R2N2 [8]	Original	**7.42**	**53.98**	9.75	48.09	0.621	4692	0.841	0.539
	DR-KFS	7.56	52.74	**9.13**	**51.58**	**0.681**	**4416**	**0.713**	**0.441**
Shape	NMR[13]	11.95	48.90	8.13	51.72	0.522	7716	0.842	0.560
Generator	SoftRas[16]	9.75	54.53	7.11	57.08	0.685	5385	**0.631**	0.471
[13,16]	DR-KFS	**8.06**	**55.24**	**5.48**	**62.02**	**0.634**	**4125**	0.741	**0.395**
IM-Net[7]	Original	7.02	66.01	1.38	69.43	0.740	3806	0.511	0.203

Note that the shape generator used in SoftRas [16] and NMR [13] are identical. We integrate DR-KFS training loss into this shape generator. Our method works well with mesh inputs. However, for voxel-based methods such as OGN/MN, we adopted a differentiable marching cubes layer (DMCL) proposed in [15] to convert voxels to meshes.

Dataset. All the tested networks are trained and evaluated on the large-scale 3D CAD model dataset, ShapeNet [5]. We use four shape categories to train and test all the networks: airplane (4,045 models), car (3,533), chair (6,778), and lamp (2,318), with an 80-20 train-test split. Input data is prepared accordingly as consumed by each one of the aforementioned networks.

Comparison Results. Table 3 shows the comparison results, where the numbers report *average* performance over all four object categories. In theory, DR-KFS can be employed for training IM-NET. Due to the limitation of our current of GPU memory, we only report the IM-NET numbers here for reference.

Our first observation is that when a network is trained using the new metric DR-KFS, it always performed better in DR-KFS, compared with the original network. Second, when trained using DR-KFS, some of the networks even

outperformed their counterparts trained with the original (object-space) losses, when the results are evaluated using CD and IoU—this is the case 4 out of 10 or 40% of the time. In particular, Matryoshka was trained using IoU, but training with DR-KFS outperformed the original network when evaluated on IoU. Next, perhaps most importantly, we see that in majority of the cases (24 out of 30 or **80%** of the time), replacing the original object-space losses using DR-KFS, the networks improved their performance in terms of visual shape similarity metrics.

4.2   Qualitative Evaluation

To understand the visual quality of the reconstructed 3D models trained using DR-KFS as the network loss, we conduct two perceptual studies (PS) on Amazon Mechanical Turk (AMT), as described below.

PS-1. In this study, we present 50 questions, each containing a single-view image and a pair of reconstructed models, one using the original loss and one using DR-KFS. Each model is rendered along two views, with one of the views aligned to the input image. We use the reconstructed models from AtlasNet [10] and Matryoshka Net [23]. Models are selected at *random* from the *entire* reconstructed set. Turkers are presented with 50 questions in a random order, with randomized orderings of the reconstructed models as well as their view-images.

PS-2. We essentially repeat PS1, but instead of selecting the models from the entire reconstructed set, we select them *randomly* from the *top-20%* of the reconstructed set, filtered based on the scores obtained using three visual similarity measures: LFD, ShapeGoogle and MvCNN.

Both PS1 and PS2 are forced-choice responses involving 80 different participants per study. For PS-1, DR-KFS metric received 71% of the total votes (4000) and for PS-2, the percentage share shot up to 86.075%, indicating a positive trend when high quality visual samples are considered for perceptual evaluation.

4.3   Design Analysis

Reconstruction using "Visual Similarity Loss vs. Image Distortion Loss". In Sect. 2, we enlisted a sampler of representative works that address the task of single-view 3D reconstruction via image supervision. We argue that simply using a multi-view pipeline is insufficient in producing visually coherent results. To support this argument, we perform an experiment where we train the Shape Generator used in SoftRas [16] (state-of-the-art work for 3D reconstruction) using two different losses: (a) Image distortion loss (per-pixel MSE), and (b) Image-similarity loss (DR-KFS framework). We combine this analysis with the effect of visual-space representation described below.

Reconstruction from "Silhouette vs. Shaded Images". Silhouette images only capture the boundary, without any surface information of the 3D shape. It is straightforward to infer that the reconstructed 3D shapes using silhouette images are likely to have bad surface quality (see Fig. 5), whereas using shaded images yields better reconstruction. To understand the differences in the reconstruction quality, we employ the shape generator used in SoftRas [16] for 3D reconstruction from either silhouette images or shaded images, with and without integrating DR-KFS pipeline.

The quantitative results for the effect of the two loss functions (visual similarity *vs.* image distortion) are shown in Table 4. DR-KFS, which measures image similarity (not distortion), wins over per-pixel MSE loss, when trained on shaded images, measured using visual shape similarity metrics such as LFD and Shape Google. Using per-pixel MSE loss on shaded images, however, is much better than using DR-KFS on silhouette images, indicating that shaded images are friendlier for the task of 3D reconstruction. This is also underscored visually in Fig. 5. This supports our claim that for a method to be *truly visual*, it should not only make use of multi-view image pipeline, but should also employ a loss that captures visual similarity and not merely image distortion.
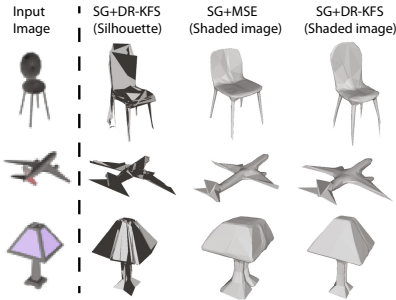

Fig. 5. 3D reconstruction results when the Shape Generator (SG) used in SoftRas [16] is trained using silhouettes vs. shaded images corresponding to an input RGB image. See Table 4 for the *average* similarity scores for the shapes shown above.

Table 4. Training a simple Shape Generator (SG) adopted in SoftRas [16] with both DR-KFS and per-pixel MSE loss. DR-KFS, which performs image feature matching, essentially incorporates visual similarity, while per-pixel MSE is merely an image distortion loss on the image pixels. The reconstructed 3D models are shown in Fig. 5.

	LFD	Shape Google
SG+DR-KFS (Silhouette)	6989	8.98
SG+MSE (Shaded Image)	5156	7.33
SG+DR-KFS (Shaded Image)	**4125**	**5.48**

Features Capturing Perceptual Differences. Given a four-legged chair with a bar between adjacent pair of legs, humans can relate to it even after vertical shifts of the bars. However, if all the bars are removed, one cannot *uniquely* associate the new chair to the original one (see Fig. 6). Cognitively, the deletion operation is more visually apparent than the vertical positional shifts of the bars. It is desired that feature descriptors encompassing a visual similarity framework should be able to capture and reflect such perceptual differences. We investigate this by calculating the image-space distances on four different features of an image (rendered-view of a 3D shape): MSE on raw image pixels, MSE on LIFT image features [34], MSE on PoI maps, and DR-KFS approach. The input image and its corresponding PoI map are shown in Fig. 6. In our experiment, vertical shifts of the bars are obtained by moving them five pixels up/down. From Table 5, we observe that DR-KFS (differentiable approach) and LIFT features (non-differentiable) are more sensitive towards the deletion operation compared to others. Moreover, image matching using DR-KFS and LIFT based local image features seem to be tolerant to small part shifts (2nd and 4th row in Table 5), while per-pixel MSE on the image I and the POI maps are quite sensitive to such relatively less perceptual changes than the deletion operation.

Table 5. Image similarity scores for the image-level operations shown in Fig. 6, using: MSE loss on image pixels (I), image features using LIFT descriptors [34], MSE loss on PoI map pixels, and DR-KFS framework. Italicized numbers (row-wise) indicate sensitivity to the respective image-level operation.
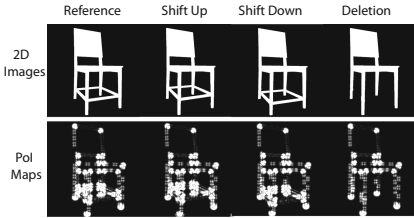
Fig. 6. Vertical positional shifts, and deletion of the leg bars. Binary images are shown on top and their corresponding PoI maps are on the bottom. Image similarity scores w.r.t the reference image for each operation using different image-level features are tabulated in Table 5.

Loss	Shift up	Shift down	Deletion
MSE (I)	0.2961	*0.3736*	0.2007
MSE(LIFT feats)	0.1174	0.1231	*0.1498*
MSE (PoI Maps)	0.0137	*0.0149*	0.0138
DR-KFS	0.0143	0.0160	*0.0256*

Fig. 7. A gallery of reconstructed 3D models obtained from AtlasNet (AN) [10], Matryoshka Net (MN) [23], Pixel2Mesh (P2M) [29] and Shape Generator (SG) [16], trained using the metrics as adopted in the respective works and by using our DR-KFS metric. Given an input image, replacing the original training loss with our DR-KFS loss results in an improvement in the visual quality of the reconstructed 3D models as shown above, and also supported by the numbers in Table 3.

5   Conclusion

We make a first step towards improving the visual quality of 3D reconstruction networks by making their training/reconstruction loss functions visual *and* driven by shape similarity rather than distortion. Overall, the new differentiable visual similarity metric we develop, DR-KFS, is shown to improve the reconstruction quality for all the tested networks (AtlasNet, Matryoshka, Pixel2Mesh, OGN, SoftRas, etc.), as judged by a variety of visual similarity measures (LFD, MVCNN, Shape Google, etc.). This is clearly a positive trend to motivate further investigation. However, the demonstrated advantage of DR-KFS is not yet "across-the-board". Our metric is still rather primitive as it does not utilize the most advanced and up-to-date tools that are available for multi-view rendering, feature extraction and learning, or image-space/perceptual assessment.

In general, the question of what the best 3D reconstruction is should depend on the application. For example, if the goal is to recognize or classify the shape, then visual similarity is more important. As well, functional understanding of an acquired shape hinges on accurate recovery of shape structures, which would support the use of visual similarity. On the other hand, if the reconstructed 3D shape is to reflect accurate physical measures, then object-space metrics should play a more prominent role.

One of the most obvious limitations of any visual similarity DR-KFS included, is that it is oblivious to errors that are *hidden* from the viewers due to occlusion. Such errors are not visible on the projected images, but they can be captured by object-space shape distances. One possibility to explore is to combine differentiable object- and image-space shape metrics as the training loss, e.g., as a weighted sum which would preserve the differentiability. This could be a good strategy to address the distortion-perception trade-off [3], but leaves the question of how to choose the weight.

References

1. Arsalan Soltani, A., Huang, H., Wu, J., Kulkarni, T.D., Tenenbaum, J.B.: Synthesizing 3D shapes via modeling multi-view depth maps and silhouettes with deep generative networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1511–1519 (2017)
2. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: speeded up robust features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006). https://doi.org/10.1007/11744023_32
3. Blau, Y., Michaeli, T.: The perception-distortion tradeoff. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6228–6237 (2018)
4. Bronstein, A.M., Bronstein, M.M., Guibas, L.J., Ovsjanikov, M.: Shape google: geometric words and expressions for invariant shape retrieval. Trans. Graph. **30**(1), 1:1–1:20 (2011)
5. Chang, A.X., et al.: Shapenet: an information-rich 3D model repository. arXiv preprint arXiv:1512.03012 (2015)

6. Chen, D.Y., Tian, X.P., Shen, Y.T., Ouhyoung, M.: On visual similarity based 3D model retrieval. In: Computer Graphics Forum, vol. 22, pp. 223–232. Wiley Online Library (2003)
7. Chen, Z., Zhang, H.: Learning implicit fields for generative shape modeling. In: CVPR (2019)
8. Choy, C.B., Xu, D., Gwak, J.Y., Chen, K., Savarese, S.: 3D-R2N2: a unified approach for single and multi-view 3D object reconstruction. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9912, pp. 628–644. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46484-8_38
9. Fan, H., Su, H., Guibas, L.J.: A point set generation network for 3d object reconstruction from a single image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 605–613 (2017)
10. Groueix, T., Fisher, M., Kim, V.G., Russell, B., Aubry, M.: Atlasnet: a papier-mâché approach to learning 3D surface generation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
11. Häne, C., Tulsiani, S., Malik, J.: Hierarchical surface prediction for 3D object reconstruction. In: 3DV (2017)
12. Joon Park, J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: DeepSDF: learning continuous signed distance functions for shape representation. In: CVPR (2019)
13. Kato, H., Ushiku, Y., Harada, T.: Neural 3D mesh renderer. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3907–3916 (2018)
14. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
15. Liao, Y., Donne, S., Geiger, A.: Deep marching cubes: learning explicit surface representations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2916–2925 (2018)
16. Liu, S., Li, T., Chen, W., Li, H.: Soft rasterizer: a differentiable renderer for image-based 3D reasoning. In: ICCV (2019)
17. Lowe, D.G., et al.: Object recognition from local scale-invariant features. In: ICCV, vol. 99, pp. 1150–1157 (1999)
18. Luo, Z., et al.: Contextdesc: local descriptor augmentation with cross-modality context. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2527–2536 (2019)
19. Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: learning 3D reconstruction in function space. In: CVPR (2019)
20. Mishchuk, A., Mishkin, D., Radenovic, F., Matas, J.: Working hard to know your neighbor's margins: local descriptor learning loss. In: Advances in Neural Information Processing Systems, pp. 4826–4837 (2017)
21. Mishkin, D., Radenovic, F., Matas, J.: Repeatability is not enough: learning affine regions via discriminability. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 284–300 (2018)
22. Pontes, J.K., Kong, C., Sridharan, S., Lucey, S., Eriksson, A., Fookes, C.: Image2Mesh: a learning framework for single image 3D reconstruction. In: Jawahar, C.V., Li, H., Mori, G., Schindler, K. (eds.) ACCV 2018. LNCS, vol. 11361, pp. 365–381. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-20887-5_23
23. Richter, S.R., Roth, S.: Matryoshka networks: predicting 3D geometry via nested shape layers. In: CVPR (2018)
24. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.R.: Orb: an efficient alternative to sift or surf. In: ICCV, vol. 11, p. 2. Citeseer (2011)

25. Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E.G.: Multi-view convolutional neural networks for 3D shape recognition. In: Proceedings of ICCV (2015)
26. Tatarchenko, M., Dosovitskiy, A., Brox, T.: Octree generating networks: Efficient convolutional architectures for high-resolution 3D outputs. In: The IEEE International Conference on Computer Vision (ICCV), October 2017
27. Tatarchenko, M., Richter, S.R., Ranftl, R., Li, Z., Koltun, V., Brox, T.: What do single-view 3D reconstruction networks learn? In: CVPR (2019)
28. Verdie, Y., Yi, K., Fua, P., Lepetit, V.: Tilde: a temporally invariant learned detector. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5279–5288 (2015)
29. Wang, N., Zhang, Y., Li, Z., Fu, Y., Liu, W., Jiang, Y.G.: Pixel2mesh: generating 3D mesh models from single RGB images. In: Proceedings of the European Conference on Computer Vision (ECCV) (2018)
30. Winder, S.A., Brown, M.: Learning local image descriptors. In: 2007 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8. IEEE (2007)
31. Wu, J., Wang, Y., Xue, T., Sun, X., Freeman, B., Tenenbaum, J.: Marrnet: 3D shape reconstruction via 2.5 D sketches. In: Advances in Neural Information Processing Systems, pp. 540–550 (2017)
32. Xu, Q., Wang, W., Ceylan, D., Mech, R., Neumann, U.: DISN: deep implicit surface network for high-quality single-view 3D reconstruction. CoRR abs/1905.10711 (2019). http://arxiv.org/abs/1905.10711
33. Yan, X., Yang, J., Yumer, E., Guo, Y., Lee, H.: Perspective transformer nets: learning single-view 3D object reconstruction without 3D supervision. In: Advances in Neural Information Processing Systems, pp. 1696–1704 (2016)
34. Yi, K.M., Trulls, E., Lepetit, V., Fua, P.: LIFT: learned invariant feature transform. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9910, pp. 467–483. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46466-4_28
35. Yu, J., Jiang, Y., Wang, Z., Cao, Z., Huang, T.: Unitbox: an advanced object detection network. In: Proceedings of the 24th ACM International Conference on Multimedia, MM 2016, pp. 516–520 (2016)