Democratising 2D Sketch to 3D Shape Retrieval Through Pivoting

Pinaki Nath Chowdhury[1,2] Ayan Kumar Bhunia[1] Aneeshan Sain[1,2] Subhadeep Koley[1,2]
Tao Xiang[1,2] Yi-Zhe Song[1,2]
[1]SketchX, CVSSP, University of Surrey, United Kingdom.
[2]iFlyTek-Surrey Joint Research Centre on Artificial Intelligence.
{p.chowdhury, a.bhunia, a.sain, s.koley, t.xiang, y.song}@surrey.ac.uk

Abstract

This paper studies the problem of 2D sketch to 3D shape retrieval, but with a focus on democratising the process. We would like this democratisation to happen on two fronts: (i) to remove the need for large-scale specifically sourced 2D sketch and 3D shape datasets, and (ii) to remove restrictions on how well the user needs to sketch and from what viewpoints. The end result is a system that is trainable using existing datasets, and once trained allows users to sketch regardless of drawing skills and without restriction on view angle. We achieve all this via a clever use of pivoting, along with novel designs that injects 3D understanding of 2D sketches into the system. We perform pivoting using two existing datasets, each from a distant research domain to the other: 2D sketch and photo pairs from the sketch-based image retrieval field (SBIR), and 3D shapes from ShapeNet. It follows that the actual feature pivoting happens on photos from the former and 2D projections from the latter. Doing this already achieves most of our democratisation challenge – the level of 2D sketch abstraction embedded in SBIR dataset offers demoralization on drawing quality, and the whole thing works without a specifically sourced 2D sketch and 3D model pair. To further achieve democratisation on sketching viewpoint, we "lift" 2D sketches to 3D space using Blind Perspective-n-Points (BPnP) that injects 3D-aware information into the sketch encoder. Results show ours achieves competitive performance compared with fully-supervised baselines, while meeting all set democratisation goals.

1. Introduction

Sketches are highly expressive [37]. This particular sketch strait has been explored to a large extend for image retrieval, especially under the fine-grained setting. The 3D literature followed a similar trend, starting with category-level retrieval of 3D shapes using sketches [45], but only very recently moved onto the fine-grained instance-level setup [58]. The very reason behind such a time lag lies with that of data collection – just as how collection of 2D sketches for images was difficult [70, 18, 66, 43], sketches for 3D models had been proven to be even harder to collect [47, 58]. In particular, since one is collecting a *2D* sketch for a *3D* model, viewpoints often need to be pre-defined [95], which arguably makes sketch collection for 3D shapes an ill-posed problem to start with – there is no one good view that caters for everyone.

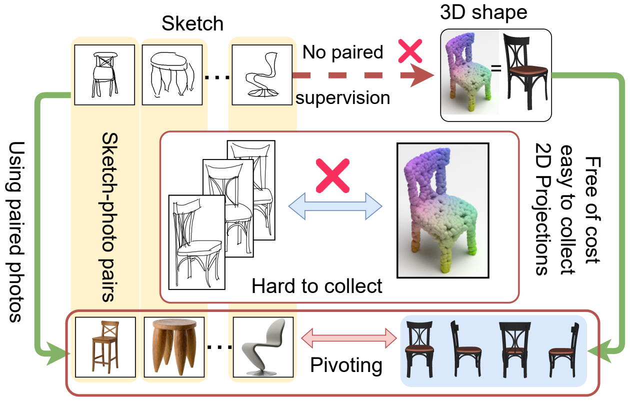

Figure 1. Collecting 2D sketches for 3D shapes is difficult since the viewpoints often needs to be pre-defined that arguably makes sketch collection for 3D shapes an ill-posed problem to start with – there is no one good view that caters for everyone. We remove the need for large-scale 2D sketch and 3D shape datasets through a clever use of pivoting across two separate domains (i) paired 2D freehand sketches from well-studied FG-SBIR literature [91], and (ii) the 3D shape domain, where we have ample data at our dispoal (e.g., ShapeNet [11]). Our pivoting happens on paired 2D photos from FG-SBIR and 2D projections from 3D shapes.

There is however a strong voice in the 3D community stating the opposite – that different to images, sketches can be freely *rendered* for 3D models, other than *collected* by hand. This is because 3D models are well-defined and without cluttered background unlike natural photos, so well-tailored non-photorealistic rendering (NPR) algorithms [26] should suffice to produce pseudo-sketches good enough for downstream tasks [19]. Indeed synthetic sketches has been explored for 3D shape retrieval [82]. Conclusion is however that they lack generalization ability to real free-hand sketches, therefore requiring an art-trained person to produce "edge-like" sketches as input.

In this paper, we set off to address all said constraints, and in effect produce for the first time a *democratised* fine-grained 2D sketch to 3D shape retrieval system. This democratisation happens on two fronts (i) we remove entirely the need for specifically sourced 2D sketch and 3D shape datasets, and (ii) we enable amateur users without art training (i.e., you and me) to still enjoy decent accuracy with their "ordinary" (more abstract) sketches.

The key innovation behind our democratisation process lies with that of pivoting [87, 41, 34, 32]. Pivoting is by now a well-explored concept in the language domain, commonly used for language translation. The idea is that by performing feature pivoting on a third shared domain, one bridges two otherwise disconnected domains (e.g., Japanese → English → French, where English is the pivoting domain). Our pivoting (see Fig. 1) also operates across two separate domains (i) the domain of paired 2D free-hand sketches, taken from the well-studied literature of fine-grained sketch-based image retrieval (FG-SBIR) [91], and (ii) the 3D shape domain, where we have ample data at our disposal (e.g., ShapeNet [11]). It follows that the pivoting factor is paired 2D photos from the former, and 2D projections of the 3D shapes from the latter. The result of this pivoting procedure is therefore a 2D sketch to 3D shape retrieval system that (i) is trainable entirely using existing datasets already proposed for diverse problem settings (FG-SBIR [91] and ShapeNet [11]), and (ii) understands sketch abstraction which is well-reflected in the sketch-photo dataset used and previously successfully modeled for FG-SBIR [52, 5].

Our democratisation challenge is mostly addressed already with just this clever use of pivoting. To inject 3D-aware knowledge into our 2D shared encoder (where pivoting is conducted), we further "lift" 2D sketch to 3D space. However, a naive 2D sketch to 3D shape generation is sub-optimal since (i) we would otherwise need 2D sketch and 3D shape pairs, thereby defeating our purpose of democratisation on specifically sourced data, and (ii) reconstructing 3D shape from sparse 2D sketches is a complex task [19] that makes our training unstable with noisy gradients. The way forward to is therefore using softer constraints. For that, we use the Blind Perspective-n-Points (BPnP) algorithm [10, 8, 9] to solves the pose and orientation between a 2D projection and 3D shapes, and later transfer this 3D-aware knowledge to sketch encoders using pivoting. Training without any 2D sketch and 3D shape pairs, the resulting framework reach an impressive SBSR performance (both category and fine-grained), reaching that of supervised (with 2D sketch - 3D shape pairs) SOTAs [58].

In summary, our contributions are: (i) we democratise fine-grained 2D sketch to 3D shape retrieval system that enable amateur users enjoy decent accuracy with their abstract sketches. (ii) The resulting framework trains without 2D sketch and 3D shape pairs via a clever use of pivoting existing FG-SBIR dataset [91] and ample 3D shapes from ShapeNet [11]. (iii) We inject 3D-aware knowledge into our 2D sketch encoder, to "lift" 2D sketch to 3D space using the BPnP algorithm [8] that solves for 2D and 3D pose and orientation. (iv) Training without any 2D sketch and 3D shape, the resulting method performs quite close to supervised SOTA [58] that use 2D sketch and 3D shape labels.

2. Related Works

Sketch for 3D Modeling: Sketching is an easy medium for people to visualize imaginary 3D objects. However, modeling 3D shape from freehand 2D sketch is an ill-posed problem [92] as any 3D object can have infinite 2D projections [1], making sketch-acquisition difficult [54]. To overcome this lack of freehand sketches, several approaches synthesise pseudo-sketches [26] for training. Despite its progress, pseudo-sketches were limited [82, 83, 92] by its inability to model deformations observed in human drawn sketches. To model these geometric distortions, hand-crafted [94] or image translation techniques [93] were applied to generate *stylised sketches*. Having uniform drawing style however, *stylised sketches* deviated significantly from their freehand counterpart. Aiming to mitigate these issues, GANs were employed, which although generated "standardized sketch" [82], removed deformations and useful drawing styles [69] observed in human-drawn freehand sketches. Similarly, domain adaptation via a domain discriminator [92] learns invariant features between synthetic and freehand sketches, but loses freehand sketch-specific information [69, 17]. In this work, instead of transforming freehand sketches to their synthetic versions, we advocate the use of pivot learning [32], where photos and 2D projection act as a pivot between freehand sketches and 3D shapes. This simultaneously avoids the use of inadequately represented pseudo-sketches and collecting large datasets of fine-grained 2D sketches for 3D shapes.

Sketch-Based 3D Shape Retrieval: Early works focused on category-level retrieval where the objective is to find a shape that belongs to the same category as the query sketch [89, 64, 44, 46]. Traditional techniques used diffusion tensor fields [90], BoF with Gabor filters [28], and part structure information [48] among others [39, 78]. This progress was motivated by the development of large category-level datasets [45, 47]. With the advent of deep learning, a common theme evolved that mapped encoded features from sketch and shape into a shared embedding space using metric learning [80, 96, 22, 86, 36, 77]. However, naively mapping 2D sketch to 3D shape is an ill-posed problem due to *view variance* – there exists multiple drastically different 2D sketches drawn from different viewpoints for the same 3D shape. To address this problem at category-level, prior works rendered multiple 2D projections [88] that are encoded using a shared CNN followed by max pool [77], or Wasserstein barycenters [86], or triplet-center loss [36]. Alternatively, siamese networks [91] learn intra-domain and cross-domain similarities from only two views [80], or compute global 3D shape descriptor using LD-SHIFT [24] or 3D-SHIFT with LLC [22]. To avoid lowering 3D shape to 2D, recent works leveraged voxels [51], point cloud [42, 59] representations, and designed 3D shape encoders like 3DCNN [84, 51] or PointNet [60, 61] accordingly.

Despite significant progress on category-level sketch-based shape retrieval (SBSR), the specialty of sketch in modelling fine-grained details [7, 16] remains largely unexplored. However, instance-level mapping not only aggravates existing problems like lack of view invariance and freehand sketch deformations [96] but is also hindered by scarcity of large-scale datasets [62]. The only work on fine-grained SBSR [58] tackles these problems by (i) curating a new dataset with $4,680$ sketch-3D pairs, (ii) rendering multiple ($n \geq 24$) 2D projections of 3D shape for an optimal match with query sketch. To avoid losing geometric details while lowering a 3D shape to 2D space, we lift [33] 2D sketches to 3D, using an auxiliary task that solves the Blind Perspective-n-Points (BPnP) problem [8, 9, 10, 12, 13]. The BPnP algorithm solves the optimisation problem of predicting the pose of a 3D object by estimating correspondence between a set of 3D points and their 2D projections. Learning this 2D-3D correspondence provides access to valuable 3D geometric constraints that help fine-grained SBSR.

Fine-Grained Sketch-Based Image Retrieval: The ability of sketches to offer inherently fine-grained visual description led to a plethora of works on fine-grained sketch-based image retrieval (FG-SBIR) [17, 6, 4, 67]. The objective is to learn pair-wise correspondence for instance-level sketch-photo matching. Aided by fine-grained sketch-photo datasets [75, 76], FG-SBIR flourished with the introduction of deep triplet-ranking model [91] for instance-level matching. This was subsequently enhanced via hybrid generative-discriminative cross-domain image generation [55], providing attention mechanism with higher order retrieval loss [76], utilising textual tags [75], or pre-training strategy [56, 2], optimal transport based retrieval [15], test-time training [67], incremental learning [3], noise-tolerant SBIR [4], invertible neural networks [17], and meta-learning [6]. Here, we show the potential of FG-SBIR to solve a long-standing problem in sketch-based 3D modelling, which aims to eliminate the dependence on pseudo-sketches [38, 53, 27, 35]. In particular, FG-SBIR can help instance-level alignment between 2D sketch and 3D model via 2D photos using a pivot-based learning paradigm [14].

3. Proposed Methodology

Overview: We aim to devise a fine-grained sketch-based shape retrieval (FG-SBSR) framework that explicitly addresses two long-standing problems (i) *freehand sketch deformation* – existing methods rely on synthesising pseudo-sketches [26] or learn feature embeddings [82] that ignore the disparity between freehand and synthetic sketches [18]. Ignoring deformations specific to freehand sketches not only limits generalisation but also deprives future research from exploring important sketch-specific information like drawing styles [69]. (ii) *view variance* – collecting large-scale instance-level 2D sketches for 3D shapes is not only difficult but ill-posed since different 2D sketches can be drawn from different viewpoints for the same 3D shape. This calls for a solution that is robust to view-variance and is simultaneously aware of deformations in a freehand sketch.

3.1. Baseline Retrieval Framework

Given a 2D sketch ($\mathbf{s} \in \mathbb{R}^{H\times W\times 3}$) and 3D geometry[1] ($\mathbf{g} \in \mathbb{R}^{N_\mathbf{g}\times 3}$) pairs, existing SBSR [59] takes a simplified approach to encode a 2D sketch using 2DCNN [74, 91] $F_{2D}(\cdot)$ and a 3D shape using 3DCNN [84, 51] or PointNet [60, 61] $F_{3D}(\cdot)$ to extract a feature map $f_\mathbf{s} = F_{2D}(\mathbf{s}) \in \mathbb{R}^d$ and $f_\mathbf{g} = F_{3D}(\mathbf{g}) \in \mathbb{R}^d$. Following [58], we use VGG-16 [74] to encode 2D sketch, and PointNet++ [61] to encode 3D coordinates from an order-invariant 3D shape point cloud. To instill discriminative knowledge, the model is trained on a cross-modal triplet objective where the cosine distance $\delta(\cdot,\cdot)$ to a 2D sketch anchor ($\mathbf{s}$) from a negative geometry ($\mathbf{g}^-$) should be increased while that from the positive geometry ($\mathbf{g}^+$) should be decreased. Training is done via triplet loss with hyperparameter $\mu > 0$ as:

$$\mathcal{L}_{trip} = \max\{0, \mu + \delta(f_\mathbf{s}, f_\mathbf{g}^+) - \delta(f_\mathbf{s}, f_\mathbf{g}^-)\} \quad (1)$$

There are some inherent limitations to this naive baseline. *Firstly*, collecting instance-level 2D sketches for a 3D shape is an ill-posed and laborious process since each 3D geometry can have multiple drastically different 2D drawings. To our best knowledge, the only instance-level paired 2D sketch – 3D shape dataset [58] contains $4,680$ one-to-one pairing across two categories ('chair' and 'lamp'). In addition, sketches in [58] are only drawn from three fixed views (0°, 30°, 75° for 'chairs', and 0°, 45°, 90° for 'lamps'). Naive solutions like generating synthetic sketches [38, 53] from 3D shapes fail to solve this dataset bottleneck due to the *freehand sketch deformation* problem – lack of robustness to human drawn sketches being unable to model real-world sketch deformations. *Secondly*, instance-level matching of a 2D sketch and 3D geometry is non-trivial since each 3D shape can be rendered into drastically different 2D projections. Existing frameworks [58] avoid this problem

[1] We use *Shape* and *Geometry* interchangeably.

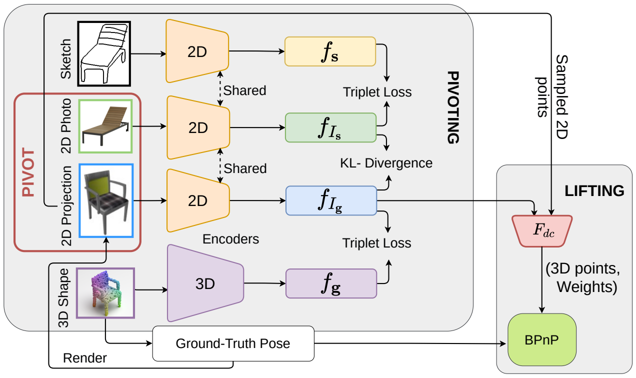

Figure 2. Our proposed method comprise of – *pivoting* and *lifting*. Pivoting bridges two disconnected datasets – sketch/photo pairs in FG-SBIR [91], and rendered 2D projections for 3D shapes [11] where photos and 2D projections act as pivot. Pivoting combines (i) triplet loss for paired sketch – photos, (ii) triplet loss for paired 3D shape – 2D projections, and (iii) distribution matching for unpaired photos – 2D projections (KL-divergence). For lifting, we combine sampled 2D points and 2D projection features to solve for pose (2D-3D correspondence) via BPnP [13]. This injects 3D-aware knowledge in the 2D encoder shared across sketch, photo, and 2D projections to simultaneously lift sketch for FG-SBSR.

by rendering $n \geq 24$ 2D projections for each 3D shape followed by an optimal matching with a 2D query sketch. This increases retrieval time by $n-$folds. *Thirdly*, existing frameworks that directly encode a 2D sketch using CNN [91, 74] and 3D shape using PointNet [60] for category-level retrieval [59] fail to generalise to instance-level mapping [58] due to the *view variance* problem that leads to an inability in capturing fine-grained 3D geometric details[2]. Therefore, this demands a further investigation on how to design a training procedure that alleviates the burden of collecting large-scale instance-level 2D sketch – 3D shape dataset while simultaneously addressing the view variance problem and allowing a faster encoding of 2D sketches and 3D shapes without the need of an elaborate pre-processing step of multiple 3D shape projections.

3.2. Pivot Learning for Sketch-Based 3D Retrieval

Pivot-based learning originated in neural machine translation (NMT) [14] for low-resource setups. It consists of three steps: (i) a source to pivot ($\mathbf{src} \rightarrow \mathbf{piv}$), and (ii) a pivot to target ($\mathbf{piv} \rightarrow \mathbf{trg}$) – parallel datasets, that are used to (iii) learn a cascaded source to target ($\mathbf{src} \rightarrow \mathbf{trg}$) translation model. The cascaded model is learned via joint training [14, 41] to maximise its likelihood. Formally,

$$P(\mathbf{trg}|\mathbf{src}) = \sum_{\mathbf{piv}} P(\mathbf{trg}|\mathbf{piv}) \cdot P(\mathbf{piv}|\mathbf{src}) \tag{2}$$

Leveraging the ease of data collection and abundance of FG-SBIR datasets [91, 75, 70], we define the first step of our pivot-based learning, i.e., $\mathbf{src} \rightarrow \mathbf{piv}$ as an instance-level retrieval between *sketch* ($\mathbf{s}$) and its *paired photo* ($I_{\mathbf{s}}$). In the second step, i.e., $\mathbf{piv} \rightarrow \mathbf{trg}$ we render multiple *2D projections* ($I_{\mathbf{g}}$) for a *3D geometry* ($\mathbf{g}$) [11] without any annotation cost and learn instance-level retrieval. We reformulate pivot-based learning objective for instance-level retrieval with hyper-parameter $\mu \geq 0$ as:

$$\begin{aligned} \mathcal{L}^{\mathbf{src}\rightarrow\mathbf{piv}}_{trip} &= \max\{0, \mu + \delta(f_{\mathbf{s}}, f^{+}_{I_{\mathbf{s}}}) - \delta(f_{\mathbf{s}}, f^{-}_{I_{\mathbf{s}}})\} \\ \mathcal{L}^{\mathbf{piv}\rightarrow\mathbf{trg}}_{trip} &= \max\{0, \mu + \delta(f_{I_{\mathbf{g}}}, f^{+}_{\mathbf{g}}) - \delta(f_{I_{\mathbf{g}}}, f^{-}_{\mathbf{g}})\} \end{aligned} \tag{3}$$

where, $f_{\mathbf{s}} = F_{2D}(\mathbf{s})$, $f_{I_{\mathbf{s}}} = F_{2D}(I_{\mathbf{s}})$, and $f_{I_{\mathbf{g}}} = F_{2D}(I_{\mathbf{g}})$, and $f_{\mathbf{g}} = F_{3D}(\mathbf{g})$. The 2D encoder F_{2D} is shared across sketch ($\mathbf{s}$), photo ($I_{\mathbf{s}}$), and 2D projections ($I_{\mathbf{g}}$). Hence, the resulting loss is defined as:

$$\mathcal{L}^{\mathbf{src}\rightarrow\mathbf{trg}}_{trip} = \mathcal{L}^{\mathbf{src}\rightarrow\mathbf{piv}}_{trip} + \mathcal{L}^{\mathbf{piv}\rightarrow\mathbf{trg}}_{trip} \tag{4}$$

Two noticeable differences exist compared to traditional pivot learning paradigms. *Firstly*, prior methods in NMT [14] worked in two scenarios where: (i) $\mathbf{src}$, $\mathbf{piv}$ and $\mathbf{trg}$ all belong to the same modality [87], (ii) $\mathbf{piv}$ belongs to either $\mathbf{src}$ [34] or $\mathbf{trg}$ [32] modality, where $\mathbf{src}$ and $\mathbf{trg}$ are in different modalities. Contrarily, our proposed pivot-learning paradigm considers $\mathbf{src}$, $\mathbf{piv}$ and $\mathbf{trg}$ all to be in three different modalities – sketch ($\mathbf{s}$), image ($I_{\mathbf{s}}, I_{\mathbf{g}}$), and 3D shape ($\mathbf{g}$). *Secondly*, pivot-based learning has been explored for generative tasks like NMT [14, 87] where maximising the joint likelihood (Eq. 2) helps generalise to $P(\mathbf{trg}|\mathbf{src})$. However, ours (Eq. 3) is a retrieval framework that learns two different metric spaces [25, 40] for instance-level mapping between sketch ($\mathbf{s}$) $\leftrightarrow$ image ($I_{\mathbf{s}}$), and image ($I_{\mathbf{g}}$) $\leftrightarrow$ geometry ($\mathbf{g}$). To facilitate the cascaded alignment across the two metric spaces allowing instance-level retrieval between sketch ($\mathbf{s}$) $\leftrightarrow$ geometry ($\mathbf{g}$), we minimise the f-divergence [31] of cosine distances $\delta(\cdot, \cdot)$ between s $\leftrightarrow$ $I_{\mathbf{s}}$ and $I_{\mathbf{g}} \leftrightarrow \mathbf{g}$. Formally,

$$\begin{aligned} \mathcal{D}^{\mathbf{src}\rightarrow\mathbf{piv}} &= \{\delta(f^{i}_{\mathbf{s}}, f^{j}_{I_{\mathbf{s}}}) : (i,j) \in [1,N]^2\} \\ \mathcal{D}^{\mathbf{piv}\rightarrow\mathbf{trg}} &= \{\delta(f^{i}_{I_{\mathbf{g}}}, f^{j}_{\mathbf{g}}) : (i,j) \in [1,M]^2\} \\ \mathcal{L}_{dist} &= D_{KL}(\mathcal{D}^{\mathbf{src}\rightarrow\mathbf{piv}} \,||\, \mathcal{D}^{\mathbf{piv}\rightarrow\mathbf{trg}}) \end{aligned} \tag{5}$$

where, N and M represent the number of paired sketch ($\mathbf{s}$) – image ($I_{\mathbf{s}}$) and 2D rendering ($I_{\mathbf{g}}$) – 3D shape ($\mathbf{g}$) pairs respectively. $\mathcal{D}_{KL}(\cdot)$ is the Kullback-Leibler divergence between two probability distributions $\mathcal{D}^{\mathbf{src}\rightarrow\mathbf{piv}}$ and $\mathcal{D}^{\mathbf{piv}\rightarrow\mathbf{trg}}$. Intuitively, we force the information radius [73] of cosine distances from sketch ($\mathbf{s}$) with image ($I_{\mathbf{s}}$) to be similar to that of 2D projections ($I_{\mathbf{g}}$) with 3D shape ($\mathbf{g}$). Hence, the total loss for our pivot-based learning is,

$$\mathcal{L}_{piv} = \mathcal{L}^{\mathbf{src}\rightarrow\mathbf{trg}}_{trip} + \mathcal{L}_{dist} \tag{6}$$

While retrieval via pivoting helps to alleviate the bottleneck of collecting instance-level 2D sketches for robustness to

[2] See supplementary for more details.

freehand sketch deformation in FG-SBSR, naively matching the encoded feature representation from 2D sketch and 3D shape using a triplet-based metric loss [68, 91] is non-trivial, as every 3D shape can be projected into largely different 2D views. Existing FG-SBSR frameworks [58] employ a cumbersome approach of rendering several ($n \geq 24$) 2D projections of each 3D shape followed by finding an optimal match with the query 2D sketch. Instead of losing information by lowering a 3D shape to 2D space, we take the alternative approach of lifting 2D sketch to 3D space as an additional auxiliary task that provides valuable geometric constraints [33] for better instance-level retrieval.

3.3. Lifting 2D Sketch to 3D Coordinates

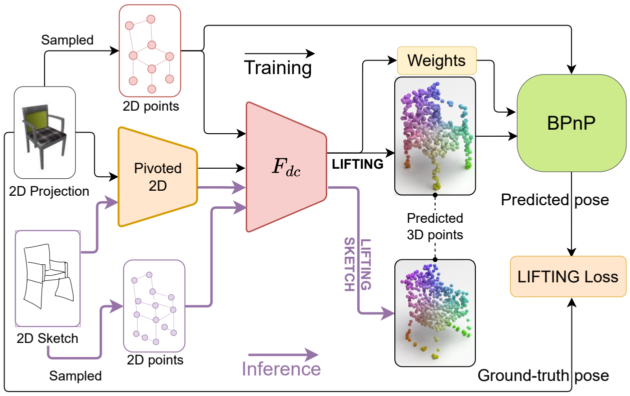

Figure 3. We inject 3D-aware knowledge into the shared 2D CNN pivoted across (sketch – photo/projections – shape) by training to predict 3D points for sampled 2D points in 2D projection. The predicted 3D points, weights, and sampled 2D points are given as input to a BPnP algorithm [13] that solves for pose of 2D projection. We compare the predicted pose with ground-truth pose of 2D projection and backpropagate this 3D-aware knowledge into the pivoted 2D CNN. Once trained, we can give input sketch to the pivoted 2D CNN along with randomly sampled points to predict their 3D coordinates. Learning 3D-aware features from sketches using the pivoted 2D encoder provides additional geometric constraints necessary for fine-grained sketch-based shape retrieval.

To provide access to valuable geometric constraints [33] necessary for fine-grained retrieval, one can "lift" 2D sketch to 3D space[3]. However, as we do not have paired 2D-3D sketch – shape instances, we inject 3D aware knowledge into the 2D encoder $F_{2D}(\cdot)$ by solving the Blind Perspective-n-Points (BPnP) algorithm between 2D projections and 3D shapes, as shown in Fig. 3. As the 2D encoder is shared across input sketch ($\mathbf{s}$), its paired photo ($I_\mathbf{s}$), and rendered 2D projections ($I_\mathbf{g}$) from 3D shape, "lifting" $I_\mathbf{g}$ (in 2D space) to 3D space helps $F_{2D}(\cdot)$ learn 3D aware knowledge, which in turn helps transform 2D sketch information ($\mathbf{s}$) to 3D space. Blind Perspective-n-Points (BPnP) algorithm [8, 9] is a pose-driven (i.e. positions and orientation) loss function, computed from a set of 3D points in shape space, and their 2D projections in image space, by learning to construct 2D-3D correspondences. Essentially, it aims to solve pose (rotation, translation) and 2D-3D correspondence simultaneously. We use BPnP as an auxiliary task only during training, without increasing inference time.

Solving the BPnP problem infuses 3D geometric knowledge into our 2D encoders $F_{2D}(\cdot)$. Given a set of (64×64) uniformly sampled 2D points $\mathbf{x} = \{x_1, x_2, \dots, x_K\}$ in $I_\mathbf{g}$, we predict their corresponding 3D points $\mathbf{z} = \{z_1, z_2, \dots, z_K\}$ and weights $\mathbf{w} = \{w_1, w_2, \dots, w_K\}$ in geometry ($\mathbf{g}$) via a dense correspondence network [13, 49] as,

$$\{\mathbf{z}, \mathbf{w}\} = F_{dc}(f_{I_\mathbf{g}}, \mathbf{x}) \tag{7}$$

where $x_i \in \mathbb{R}^2$, $z_i \in \mathbb{R}^3$, $w_i \in \mathbb{R}^2$, and $f_{I_\mathbf{g}} = F_{2D}(I_\mathbf{g})$. Directly regressing on the predicted $\mathbf{z}$ with ground-truth 3D points to learn 2D-3D correspondence does not leverage geometric priors [13]. Hence, we supervise by solving pose using geometry-based BPnP algorithms to have stable generalisation [13, 12]. The BPnP algorithm takes sampled 2D points ($\mathbf{x}$), predicted 3D points ($\mathbf{z}$) and weights ($\mathbf{w}$) as input to solve for an optimal pose $y = \{R, t\}$ (rotation matrix $R \in \mathbb{R}^{3\times3}$ and translation vector $t \in \mathbb{R}^{3\times1}$). Hence, instead of supervising the output of F_{dc}, we add a BPnP module to compute loss on predicted pose (y). BPnP minimises the projection error given by the following optimisation:

$$y_{pred} = \arg\min_{y} \frac{1}{2} \sum_{i=1}^{N} || \underbrace{w_i \circ (\pi(Rz_i + t) - x_i)}_{\Phi_i(y)} ||^2 \tag{8}$$

where $\pi(\cdot)$ represents the 2D projection function with intrinsic camera parameters, and $\circ$ represents element-wise product. The projection error in Eq. 8 is analogous to least squares error $||y_{pred} - y_{gt}||^2$ that leads to non-unique solutions[4] [12, 13]. To overcome this ambiguity [71, 50] in estimating pose y_{pred} from 2D projections, following the definition in [13], we reformulate Eq. 8 as a probability density estimation problem as:

$$p(\mathbf{z}, \mathbf{x}, \mathbf{w}|y) = p(\theta|y) = \exp\left(-\frac{1}{2}\sum_{i=1}^{N} ||\Phi_i(y)||^2\right) \tag{9}$$
$$\text{where } \Phi_i(y) = w_i \circ (\pi(Rz_i + t) - x_i)$$

Hence, the new training objective matches the predicted pose distribution $p(y|\theta)$ with ground-truth pose distribution $p(y_{gt})$ using KL-divergence as,

$$\begin{aligned} \mathcal{L}_{KL} &= D_{KL}(p(y_{gt}) \,||\, p(y|\theta)) = \\ &- \int p(y_{gt}) \log p(\theta|y) dy + \log \int p(\theta|y) dy \end{aligned} \tag{10}$$

[3] Unlike [33], by "lifting" we do not reconstruct a 3D sketch from 2D, but inject 3D aware knowledge into the 2D sketch encoder.

[4] Why non-unique? Please see supplementary for more details.

While we model the probability distribution of the predicted pose using $p(y|\theta)$, its ground-truth (GT) pose distribution $p(y_{gt})$ corresponds to one unique pose. We thus choose the ground-truth $p(y_{gt})$ as a narrow Dirac delta distribution as,

$$\mathcal{L}_{KL} = \underbrace{\frac{1}{2}\sum_{i=1}^{N}||\Phi_i(y_{gt})||^2}_{\text{reprojection at GT pose}} + \underbrace{\log\int \exp -\frac{1}{2}\sum_{i=1}^{N}||\Phi_i(y)||^2 dy}_{\text{reprojection at predicted pose}} \tag{11}$$

The first term in Eq. 11 measures the uncertainty of projection error given the ground-truth pose y_{gt}, whereas the second term represents the variance of projection error over the predicted pose y. Lowering the variance term helps increase discriminative ability, thus lowering pose ambiguity. To solve the variance of projection error, we use Monte-Carlo approach following the Adaptive Multiple Importance Sampling (AMIS) algorithm [20] as,

$$\begin{aligned}\mathcal{L}_{var} = \log\int \exp -\frac{1}{2}\sum_{i=1}^{N}||\Phi_i(y)||^2 dy \approx \\ \log\frac{1}{K}\sum_{j=1}^{K}\frac{1}{q(y_j)}\left\{\exp -\frac{1}{2}\sum_{i=i}^{N}||\Phi_i(y_j)||^2\right\}\end{aligned} \tag{12}$$

where $q(y_j)$ is the proposal distribution for position and orientation. The choice of $q(y_j)$ strongly affects the numerical stability. Following existing literature [79, 13], for position we choose 3 DoF multivariate t-Distributions, and for orientation we use the angular central Gaussian distribution. The final loss used in BPnP regularisation loss becomes[5],

$$\mathcal{L}_{reg} = \frac{1}{2}\sum_{i=1}^{N}||\Phi_i(y_{gt})||^2 + \mathcal{L}_{var} \tag{13}$$

Our final training objective, combines a pivot-based retrieval learning $\mathcal{L}_{piv}$ (in Eq. 6) with a regularisation loss $\mathcal{L}_{reg}$ (in Eq. 13) over a weighting hyperparameter λ as,

$$\mathcal{L}_{tot} = \mathcal{L}_{piv} + \lambda \cdot \mathcal{L}_{reg} \tag{14}$$

4. Experiments

Datasets: Being a recently explored task, there exists only one dataset for FG-SBSR by Qi *et al.* [58] that comprises 1005 and 555 sketch-3D shape quadruplets of 'chairs' and 'lamps'. Each quadruplet contains three sketches from different views (0°, 30°, 75° for 'chairs', and 0°, 45°, 90° for 'lamps') and one 3D shape. Following [58], we use 804 and 444 quadruplets respectively (i.e., 80%) for training, and the rest for testing. Our pivot-based training requires two datasets: ($\mathbf{src} \rightarrow \mathbf{piv}$) and ($\mathbf{piv} \rightarrow \mathbf{trg}$). For source to pivot, we use QMUL-Chair-V2 [91] consisting of 1275/725 sketched chairs and 300/100 photos for training/testing respectively. Pivot to target can be freely generated by 2D projections ($n = 360$) of 3D shape models in Qi *et al.* [58]. Although our focus is FG-SBSR, we also test for generalisability of our proposed method on category-level SBSR tasks. In particular, we evaluate on SHREC'13 [45] and SHREC'14 [47] sketch track benchmark datasets to compare with existing state-of-the-art category-level SBSR methods [88, 86]. SHREC'13 [45] includes $7,200$ hand-drawn sketches and $1,258$ 3D shapes from 90 categories. SHREC'14 [47] extends SHREC'13 [45] with $13,680$ hand-drawn sketches and $8,987$ 3D shapes from 171 categories. On average, both datasets have 80 sketches for each category divided into 50 for training and 30 for testing [86].

Implementation Details: Our model is implemented in PyTorch on a 11GB Nvidia RTX 2080-Ti GPU. We train our model for 200 epochs using Adam optimiser with learning rate 0.0001, batch size 8, accumulating gradients over 8 steps. The margin value for triplet loss μ is set to 0.3. To reduce computational overhead, we use 512 monte carlo samples and 4 Adaptive Multiple Instance Sampling (AMIS) iterations. For $F_{2D}(\cdot)$ we use VGG-16 [74] whereas $F_{dc}(\cdot)$ consists of a rotation and translation head described in [49]. The learned rotation and frozen translation head predicts (64×64) 3D points $\mathbf{z} \in \mathbb{R}^{64\times64\times3}$ and weights $\mathbf{w} \in \mathbb{R}^{64\times64\times2}$. We set the weighting factor λ balancing $\mathcal{L}_{piv}$ and $\mathcal{L}_{reg}$ as 0.1. Finally, Levenberg-Marquardt (LM) PnP solver – a robust variation of the Gauss-Newton algorithm, with Huber kernel [72] and adaptive threshold computes pose given 2D-3D correspondence[6].

Evaluation Metric: In line with FG-SBSR research [91], we use Acc.@q, i.e. percentage of sketches having true matched photo in the top-q list. For category-level SBSR, we follow the widely-adopted metrics such as: (i) Nearest Neighbor (NN) computes percentage of 3D shapes in the top-1 list. (ii) First/Second Tier (FT/ST) measures the percentage of correctly labeled 3D shapes in the top $(C-1)$ or $2(C-1)$ list, where C is the number of 3D samples in query's class. (iii) E-Measure (E) combines precision and recall into a single number. (iv) Discounted Cumulated Gain (DCG) weighs the correctly retrieved results towards the beginning of the ranked list more than those towards its end. (v) Mean Average Precision (mAP) computes average precision for each query sketch. We use the source code in [62] to compute category-level SBSR metrics.

4.1. Competitors

We compare against (i) state-of-the-art (SOTA) that improve 3D shape representation. These methods are divided into (a) encode 3D shape information without 2D projections [61, 29], i.e., non-projection based 3D encoders. **FG-PointNet** use PointNet++ [60] for 3D shape and VGG-16 [74] for 2D shape encoding respectively. Training progresses over triplet loss [91]. **FG-Spherical** replaces Point-

[5] For a detailed tutorial, see supplementary material.

[6] See supplementary for a detailed PyTorch-like code

Net++ with Spherical CNN [29]. (b) capture 3D shape information via multiple 2D projections [77, 23, 58] to overcome view variance between 2D sketch and 3D shape. **MVCNN** [77] projects 3D shape into ($n = 24$) 2D views. Each view is separately encoded with a shared 2D encoder followed by max-pooling to get the resulting 3D shape feature. **MVAvg** [23] follows *MVCNN* but replaces max-pooling with average-pooling. **MVAttn** by Qi *et al.* [58] improves upon *MVCNN* and *MV-Avg* via cross-modal view attention to dynamically determine relevant 2D projections given a 2D sketch for a weighted sum. (ii) Focusing on improving the 2D sketch representation, we adopt techniques from sketch-based 3D modeling literature [93, 92, 82]. A better 2D sketch and 3D shape mapping is learned using non-photo realistic renderings [38, 53, 27] to generate large-scale datasets of synthetic sketch and 3D shape. To achieve robustness to freehand sketch deformations, **GANSketch** [82] transforms a freehand sketch to the synthesised sketch domain. We train it using synthetic sketch with triplet loss. For evaluation, a pretrained standardized module is used to transform freehand sketch to synthetic domain for retrieval. **AdaptSketch** [92] employs a domain discriminator that trains our 2D sketch encoder adversarially for robustness to synthetic and freehand sketches. While the retrieval model is trained using large-scale synthetic sketches and triplet loss, the domain discriminator additionally requires a small set of freehand sketches. **StylisedSketch** [93] applies a set of random global and local deformations to each stroke in a synthesised sketch. The resulting sketch is given to a CNN [81] that consolidates and simplifies to give style-unified sketch. Training progresses using style-unified sketch with triplet loss. (iii) An alternative to improving 2D sketch encoder is injecting 3D aware knowledge for fine-grained SBSR via 2D sketch to 3D shape reconstruction. **SDFSketch** employs DeepSDF [57] as an auxiliary task only during training to generate 3D shape from a 2D sketch representation. The reconstruction module predicts signed distance field [21] trained using $L1$ loss. **PSGNSketch** follows *SDFSketch* but uses PSGN [30] to predict a point-based shape representation trained using chamfer distance [85] as reconstruction loss.

4.2. Evaluation on Fine-Grained SBSR

We evaluate on 'chairs' and 'lamps' [58]. Our setup includes: (i) *zero-shot* that use pivoting to train with fine-grained sketch-photo instances in QMUL-Chair-V2 [91] and freely available 3D shapes – 2D projections in Qi *et al.* [58]. (ii) Although not our main goal, we further report the *upper-bound/all-shot* that fine-tunes the zero-shot model using paired 2D sketch – 3D shape in Qi *et al.* [58].

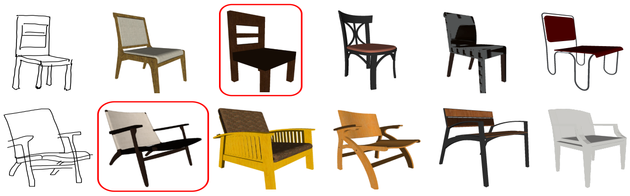

Figure 4. Qualitative top-5 fine-grained retrieved results on Qi *et al.* [58] using our proposed method. RED denotes GT 3D shape.

Performance Analysis: From Table. 1, we observe: (i) *2D Enc.* perform lower than *3D Enc.*. This shows training with synthetic sketches does not generalise to freehand sketch deformations. (ii) Injecting 3D aware knowledge in *Lift* improves performance over *2D Enc.* and is competitive with *3D Enc.*. It confirms our intuition that instead of lowering 3D shapes to 2D space with multiple 2D projections, one can "lift" 2D sketches to 3D space for improved fine-grained matching. (iii) *Ours (zero-shot)* is competitive with existing SOTA *3D Enc.* without using any labelled training data. (iv) *zero-shot* performance is lower in 'lamps' than in 'chairs'. This is because the training data (**src** → **piv**) [91] used in pivoting only contains 'chairs' and not 'lamps'. See supplementary for more details on 'lamps'. (v) Combining pivot-based pretraining along with 3D aware knowledge, injected using BPnP [10] outperforms existing methods without using expensive 2D projections ($n = 24$). (vi) Fig. 4 visualises top-10 retrievals on Qi *et al.* [58].

Table 1. Comparative fine-grained SBSR results on Qi *et al.* [58]. Following section 4.1, we compare across groups of methods (i) **3D Enc.**: improves 3D encoders $F_{3D}(\cdot)$, (ii) **2D Enc.**: improves 2D encoders $F_{2D}(\cdot)$, and (iii) **Lift**: injects 3D aware knowledge into $F_{2D}(\cdot)$ using auxiliary 2D sketch to 3D reconstruction task.

	Method	Chairs		Lamps	
		Acc.@1	Acc.@5	Acc.@1	Acc.@5
3D Enc.	FGPointNet [61]	0.50	4.48	0.90	3.60
	FGSperical [29]	11.77	40.13	12.61	38.44
	MVCNN [77]	47.60	81.26	49.25	83.48
	MVAvg [23]	45.12	77.94	48.56	80.11
	MVAttn [58]	56.72	87.06	57.66	87.39
2D Enc.	GANSketch [82]	37.53	65.92	35.91	66.10
	AdaptSketch [92]	34.66	65.66	35.47	65.46
	StylisedSketch [93]	30.21	64.76	31.34	63.75
Lift	SDFSketch [57]	56.91	87.43	58.11	87.40
	PSGNSketch [30]	57.42	88.10	58.35	87.76
Ours	**zero-shot**	**55.79**	**86.12**	**21.63**	**57.66**
	upper-limit/all-shot	**58.53**	**89.67**	**58.74**	**87.95**

4.3. Evaluation on Category-Level SBSR

Although we focus on fine-grained matching between 2D sketch and 3D shapes, to evaluate the generalisability of our method, we evaluate on category-level SBSR benchmark datasets SHREC'13 [45] and SHREC'14 [47]. (i) For *zero-shot*, following [88] we split seen/unseen categories into $79/11$ for SHREC'13 [45], and $151/20$ for SHREC'14 [47]. (ii) To encourage future work (not our goal), we report *upper-limit/all-shot* with paired 2D sketch – 3D shapes.

Performance Analysis: From Tab. 2, we observe: (i) The performance gap between *2D Enc.* and *3D Enc.* is slightly

Table 2. Comparative category-level SBSR results on benchmark SHREC'13 [45] and SHREC'14 [47] datasets.

	Method	SHREC'13 [45]						SHREC'14 [47]					
		NN	FT	ST	E	DCG	mAP	NN	FT	ST	E	DCG	mAP
3D Enc.	FGPointNet [61]	82.3	82.8	86.0	40.3	88.4	84.3	80.4	74.9	81.3	39.5	87.0	78.0
	FGSperical [29]	80.4	80.9	83.6	40.1	87.3	82.9	77.9	72.6	79.9	39.1	86.3	76.8
	MVCNN [77]	76.3	78.7	84.9	39.2	85.4	80.7	58.5	45.5	53.9	27.5	66.6	47.7
	MVAvg [23]	71.2	72.5	78.5	36.9	81.4	75.2	40.3	37.8	45.5	23.6	58.1	40.1
	MVAttn [58]	83.4	**85.4**	90.1	41.8	90.1	87.2	78.7	81.2	84.9	41.9	88.2	83.1
2D Enc.	GANSketch [82]	67.5	68.3	72.8	36.3	77.4	68.6	30.4	28.9	36.0	18.9	52.4	30.6
	AdaptSketch [92]	64.3	63.1	71.6	34.4	76.3	67.1	27.9	28.3	34.9	18.5	50.1	29.4
	StylisedSketch [93]	62.6	63.4	68.8	35.8	74.3	64.7	23.5	25.6	31.7	15.4	44.9	26.1
Lift	SDFSketch [57]	66.7	67.1	70.5	36.0	77.1	68.2	28.5	28.3	35.1	18.6	50.5	29.9
	PSGNSketch [30]	67.4	68.1	72.4	36.0	76.9	68.1	29.6	28.5	35.3	18.7	51.2	30.1
Ours	**zero-shot**	**55.3**	**49.5**	**67.5**	**37.4**	**70.1**	**55.7**	**43.1**	**35.9**	**46.8**	**19.6**	**59.5**	**37.6**
	upper-limit/all-shot	**84.9**	**85.4**	**90.7**	**42.3**	**90.7**	**87.6**	**80.1**	**81.3**	**86.2**	**43.3**	**88.9**	**83.7**

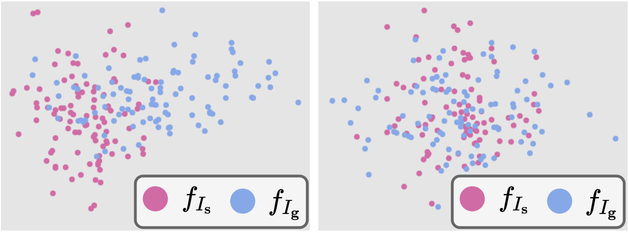

Figure 5. Visualising the importance of $\mathcal{L}_{dist}$ on pivot-based learning with triplet loss. Using $\mathcal{L}_{dist}$, the t-SNE plots of f_{I_s} and f_{I_g} are more closely aligned. This helps pivoting $\mathbf{s}$ to $\mathbf{g}$.

lower than its fine-grained counterpart (in Tab. 1). This indicates that the issue of freehand sketch deformation is more pronounced in fine-grained setup. (ii) Injecting 3D aware knowledge in *Lift* fails to improve performance over *3D Enc*. Reconstructing 3D shape from category-level 2D sketches is a difficult problem thus leading to unstable generalisation. (iii) Instead of generating 3D shape our *Proposed Method* only predicts 3D coordinates of visible 2D points. This softer constraint [13] helps in better generalisation to category-level retrieval, making our *Proposed Method* competitive with existing SOTAs.

Table 3. Ablative study on 'chairs' in Qi *et al*. [58]. *Piv* denotes pivot-based learning and *all-shot* represents the use of paired 2D sketch and 3D shape from train-set in [58].

Piv	$\mathcal{L}_{dist}$	BPnP	all-shot	Acc.@1	Acc.@5
✗	✗	✗	✓	10.5	38.7
✓	✗	✗	✓	48.7	80.7
✓	✗	✗	✗	46.5	78.5
✓	✓	✗	✗	49.3	82.2
✓	✓	✓	✗	55.8	86.1
✓	✓	✓	✓	58.5	89.7

4.4. Ablation

Importance of Pivoting: We remove constraints like $\mathcal{L}_{dist}$ (w/o – $\mathcal{L}_{dist}$), BPnP solver (w/o – BPnP), and compare (Table. 3) our baseline (w/o – *Piv*, w – *all-shot*) against using pivoting as pretraining (w–*Piv*, w–*all-shot*). We observe 38.2/42.0 rise in Acc.@1/Acc.@5 while pivoting.

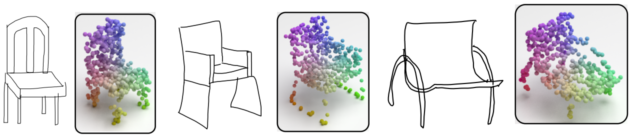

Figure 6. Visualising predicted 3D points $\mathbf{z}$ from sketch (Eq. 7).

Importance of enforcing $\mathcal{L}_{dist}$: To evaluate contribution of $\mathcal{L}_{dist}$, we compare *zero-shot* performance using only pivoting (w – *Piv*, w/o – $\mathcal{L}_{dist}$) against $\mathcal{L}_{dist}$ with pivoting (w – *Piv*, w – $\mathcal{L}_{dist}$). Tab. 3 shows 2.8/3.7 improvement in Acc.@1/Acc.@5 when using $\mathcal{L}_{dist}$. Additionally, t-SNE plot in Fig. 5 shows better alignment between photos paired with sketches ($I_{\mathbf{s}}$) and 2D projections ($I_{\mathbf{g}}$) of 3D shapes when using $\mathcal{L}_{dist}$. This alignment helps map source 2D sketch ($\mathbf{s}$) with target 3D shape ($\mathbf{g}$).

Importance of solving BPnP: From Tab. 3, we measure the effect of adding the auxiliary BPnP solve during training. Comparing pivoting with $\mathcal{L}_{dist}$ (w – *Piv*, w – $\mathcal{L}_{dist}$, w/o – BPnP) against adding a BPnP auxiliary training loss (w – *Piv*, w – $\mathcal{L}_{dist}$, w – BPnP), we observe 6.5/3.9 improvement in Acc.@1/Acc.@5. Furthermore, Fig. 6 visualises the predicted 3D points from 2D sketch via our dense correspondence network $F_{dc}(\cdot)$ in Eq. 7.

Limitations and Future Improvements: From our ablation study (Tab. 3), we observe that even after using $\mathcal{L}_{dist}$, pivoting, and BPnP solver, fine-tuning our proposed method on labelled 2D sketch and 3D shape pairs improves by 2.7/3.6 in Acc.@1/Acc.@5. This gap between *zero-shot* and *all-shot* shows an important limitation of all FG-SBSR methods – despite being an ill-posed problem, collecting 2D sketch for 3D shapes (*all-shot*) is still necessary to outperform its *zero-shot* setup (using unpaired 2D sketch and 3D shapes). Our future work would thus aim to overcome this *zero-shot* gap without the need of any FG-SBSR training data. Our intuition is that adapting CLIP [63] for generalised cross-category FG-SBIR along with pivoting and BPnP solver can finally resolve this important limitation.

Failure Cases: As mentioned in Sec. 4.2, the performance

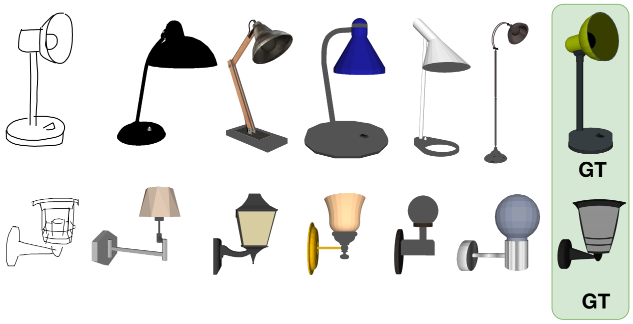

Figure 7. Failure cases for Fine-Grained SBSR for 'Lamps' [58].

in 'lamps' is lower than 'chairs', as no 2D sketch/photo 'lamps' dataset was available for training via pivot learning (unlike Chair-V2 [91] for 'chairs'). We thus attempted the harder problem of training on 'chairs' and evaluating on 'lamps'. This led to a significant performance drop for fine-grained SBSR on 'lamps'. Fig. 7 shows some failure cases for fine-grained SBSR on 'lamps'. Future works can exploit the zero-shot generalisation of CLIP [63] for cross-category fine-grained retrieval [65] and help remove the need to collect a 2D sketch/photo datasets for pivot learning.

5. Conclusion

In this paper, we scrutinise two important bottlenecks in fine-grained sketch-based shape retrieval: (i) view variance – collecting 2D sketches for 3D shapes is difficult since the viewpoints often needs to be pre-defined that arguably makes sketch collection for 3D shapes an ill-posed problem to start with. There is no one good view that caters for everyone. (ii) freehand sketch deformation – existing methods that rely on synthesising pseudo-sketches ignore the disparity between freehand and synthetic sketches. For this, we overcome the need of collecting large-scale 2D sketches for 3D shapes by first introducing pivot-based learning for 3D shape retrieval. Next, we inject 3D-aware knowledge in our 2D sketch encoder to "lift" 2D sketch to 3D space that provides additional geometric cues to improved fine-grained 2D sketch – 3D shape matching. Empirical evidence shows remarkable performance gain, even in *zero-shot* setups.

References

[1] Harry G. Barrow and Jay M. Tenenbaum. Interpreting Line Drawings as Three-Dimensional Surfaces. In *AAAI*, 1981. 2

[2] Ayan Kumar Bhunia, Pinaki Nath Chowdhury, Yongxin Yang, Timothy M. Hospedales, Tao Xiang, and Yi-Zhe Song. Vectorization and Rasterization: Self-Supervised Learning for Sketch and Handwriting. In *CVPR*, 2021. 3

[3] Ayan Kumar Bhunia, Viswanatha Reddy Gajjala, Subhadeep Koley, Rohit Kundu, Aneeshan Sain, Tao Xiang, and Yi-Zhe Song. Doodle It Yourself: Class Incremental Learning by Drawing a Few Sketches. In *CVPR*, 2022. 3

[4] Ayan Kumar Bhunia, Subhadeep Koley, Abdullah Faiz Ur Rahman Khilji, Aneeshan Sain, Pinaki Nath Chowdhury, Tao Xiang, and Yi-Zhe Song. Sketching without Worrying: Noise-Tolerant Sketch-Based Image Retrieval. In *CVPR*, 2022. 3

[5] Ayan Kumar Bhunia, Subhadeep Koley, Amandeep Kumar, Aneeshan Sain, Pinaki Nath Chowdhury, Tao Xiang, and Yi-Zhe Song. Sketch2Saliency: Learning to Detect Salient Objects from Human Drawings. In *CVPR*, 2023. 2

[6] Ayan Kumar Bhunia, Aneeshan Sain, Parth Hiren Shah, Animesh Gupta, Pinaki Nath Chowdhury, Tao Xiang, and Yi-Zhe Song. Adaptive Fine-Grained Sketch-Based Image Retrieval. In *ECCV*, 2022. 3

[7] Ayan Kumar Bhunia, Yongxin Yang, Timothy M. Hospedales, Tao Xiang, and Yi-Zhe Song. Sketch Less for More: On-the-Fly Fine-Grained Sketch Based Image Retrieval. In *CVPR*, 2020. 3

[8] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. Dsac - Differentiable RANSAC for Camera Localization. In *CVPR*, 2017. 2, 3, 5

[9] Eric Brachmann and Carsten Rother. Learning Less is More - 6D Camera Localization via 3D Surface Regression. In *CVPR*, 2018. 2, 3, 5

[10] Dylan Campbell, Liu Liu, and Stephen Gould. Solving the Blind Perspective-n-Point Problem End-to-End with Robust Differentiable Geometric Optimization. In *ECCV*, 2020. 2, 3, 7

[11] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. Shapenet: An Information-Rich 3D Model Repository. *arXiv preprint arXiv:1512.03012*, 2015. 1, 2, 4

[12] Bo Chen, Alvaro Parra, Jiewei Cao, Nan Li, and Tat-Jun Chin. End-to-End Learnable Geometric Vision by Backpropagating PnP Optimization. In *CVPR*, 2020. 3, 5

[13] Hansheng Chen, Pichao Wang, Fan Wang, Wei Tian, Lu Xiong, and Hao Li. EPro-PnP: Generalized End-to-End Probabilistic Perspective-n-Points for Monocular Object Pose Estimation. In *CVPR*, 2022. 3, 4, 5, 6, 8

[14] Yong Chen, Qian Yang, Yang Liu, Maosong Sun, and Wei Xu. Joint Training for Pivot-based Neural Machine Translation. *IJCAI*, 2017. 3, 4

[15] Pinaki Nath Chowdhury, Ayan Kumar Bhunia, Viswanatha Reddy Gajjala, Aneeshan Sain, Tao Xiang, and Yi-Zhe Song. Partially Does It: Towards Scene-Level FG-SBIR with Partial Input. In *CVPR*, 2022. 3

[16] Pinaki Nath Chowdhury, Ayan Kumar Bhunia, Aneeshan Sain, Subhadeep Koley, Tao Xiang, and Yi-Zhe Song. What Can Human Sketches Do for Object Detection? In *CVPR*, 2023. 3

[17] Pinaki Nath Chowdhury, Ayan Kumar Bhunia, Xiang Tao, and Yi-Zhe Song. SceneTrilogy: On Scene Sketches and its Relationship with Text and Photo. In *CVPR*, 2023. 2, 3

[18] Pinaki Nath Chowdhury, Aneeshan Sain, Tao Xiang, Yulia Gryaditskaya, and Yi-Zhe Song. FS-COCO: Towards Understanding of Freehand Sketches of Common Objects in Context. In *ECCV*, 2022. 1, 3

[19] Pinaki Nath Chowdhury, Tuanfeng Wang, Duygu Ceylan, Yi-Zhe Song, and Yulia Gryaditskaya. Garment Ideation: Iterative view-aware sketch-based garment modeling. In *3DV*, 2022. 2

[20] Jean-Marie Cornuet, Jean-Michel Marin, Antonietta Mira, and Christian P. Robert. Adaptive Multiple Importance Sampling. *Scandinavian Journal of Statistics*, 2012. 6

[21] Angela Dai, Charles Ruizhongtai Qi, and Matthias Nießner. Shape Completion using 3D-Encoder-Predictor CNNs and Shape Synthesis. In *CVPR*, 2017. 7

[22] Guoxian Dai, Jin Xie, Fan Zhu, and Yi Fang. Deep Correlated Metric Learning for Sketch-Based 3D Shape Retrieval. In *AAAI*, 2017. 2, 3

[23] Weidong Dai and Shuang Liang. Cross-Modal Guidance Network For Sketch-Based 3D Shape Retrieval. In *ICME*, 2020. 7, 8

[24] Tal Darom and Yosi Keller. Scale-Invariant Features for 3D Mesh Models. *TIP*, 2012. 3

[25] Jason V. Davis, Brian Kulis, Prateek Jain, Suvrit Sra, and Inderjit S. Dhillon. Information-Theoretic Metric Learning. In *ICML*, 2007. 4

[26] Doug DeCarlo, Adam Finkelstein, Szymon Rusinkiewicz, and Anthony Santella. Suggestive Contours for Conveying Shape. *ACM Trans. Graph*, 2003. 1, 2, 3

[27] Johanna Delanoy, Mathieu Aubry, Phillip Isola, Alexei A Efros, and Adrien Bousseau. 3D Sketching using Multi-View Deep Volumetric Prediction. *CGIT*, 2018. 3, 7

[28] Mathias Eitz, Ronald Richter, Tamy Boubekeur, Kristian Hildebrand, and Marc Alexa. Sketch-based Shape Retrieval. *ACM Trans. Graph*, 2012. 2

[29] Carlos Esteves, Christine Allen-Blanchette, Ameesh Makadia, and Kostas Daniilidis. Learning SO(3) Equivariant Representations with Spherical CNNs. In *ECCV*, 2018. 6, 7, 8

[30] Haoqiang Fan, Hao Su, and Leonidas Guibas. A Point Set Generation Network for 3D Object Reconstruction from a Single Image. In *CVPR*, 2017. 7, 8

[31] Dario García-García and Robert C. Williamson. Divergences and Risks for Multiclass Experiments. In *COLT*, 2012. 4

[32] Spandana Gella, Rico Sennrich, Frank Keller, and Mirella Lapata. Image Pivoting for Learning Multilingual Multimodal Representations. In *EMNLP*, 2017. 2, 4

[33] Yulia Gryaditskaya, Felix Hähnlein, Chenxi Liu, Alla Sheffer, and Adrien Bousseau. Lifting Freehand Concept Sketches into 3D. *ACM Trans. Graph*, 2020. 3, 5

[34] Jiuxiang Gu, Shafiq Joty, Jianfei Cai, and Gang Wang. Unpaired Image Captioning by Language Pivoting. In *ECCV*, 2018. 2, 4

[35] Benoit Guillard, Edoardo Remelli, Pierre Yvernay, and Pascal Fua. Sketch2Mesh: Reconstructing and Editing 3D Shapes from Sketches. In *ICCV*, 2021. 3

[36] Xinwei He, Yang Zhou, Zhichao Zhou, Song Bai, and Xiang Bai. Triplet-Center Loss for Multi-View 3D Object Retrieval. In *CVPR*, 2018. 2, 3

[37] Aaron Hertzmann. Why Do Line Drawings Work? A Realism Hypothesis. *Perception*, 2020. 1

[38] Haibin Huang, Evangelos Kalogerakis, Ersin Yumer, and Radomir Mech. Shape Synthesis from Sketches via Procedural Models and Convolutional Networks. *IEEE TVCG*, 2016. 3, 7

[39] Anastasia Ioannidou, Elisavet Chatzilari, Spiros Nikolopoulos, and Ioannis Kompatsiaris. Deep Learning Advances in Computer Vision with 3D Data: A Survey. *ACM Computing Surveys*, 2017. 2

[40] Dor Kedem, Stephen Tyree, Fei Sha, Gert Lanckriet, and Kilian Q. Weinberger. Non-linear Metric Learning. In *NeurIPS*, 2012. 4

[41] Yunsu Kim, Petre Petrov, Pavel Petrushkov, Shahram Khadivi, and Hermann Ney. Pivot-based Transfer Learning for Neural Machine Translation between Non-English Languages. In *EMNLP*, 2019. 2, 4

[42] Roman Klokov and Victor Lempitsky. Escape From Cells: Deep KD-Networks for The Recognition of 3D Point Cloud Models. In *ICCV*, 2017. 3

[43] Subhadeep Koley, Ayan Kumar Bhunia, Aneeshan Sain, Pinaki Nath Chowdhury, Tao Xiang, and Yi-Zhe Song. Piature that Sketch: Photorealistic Image Generation from Abstract Sketches. In *CVPR*, 2023. 1

[44] Bo Li and Henry Johan. Sketch-based 3D Model Retrieval by Incorporating 2D-3D Alignment. *MTAP*, 2013. 2

[45] Bo Li, Yijuan Lu, Afzal Godil, Tobias Schreck, Masaki Aono, Henry Johan, Jose M. Saavedra, and Shoki Tashiro. SHREC'13 track: large scale sketch-based 3D shape retrieval. In *3DOR*, 2013. 1, 2, 6, 7, 8

[46] Bo Li, Yijuan Lu, Henry Johan, and Ribel Fares. Sketch-based 3D Model Retrieval Utilizing Adaptive View Clustering and Semantic Information. *MTAP*, 2017. 2

[47] Bo Li, Y. Lu, Chen-Feng Li, Afzal A. Godil, Tobias Schreck, Aono, Martin Burtscher, Hongbo Fu, Takahiko Furuya, H. Johan, J. Liu, Ryutarou Ohbuchi, A. Tatsuma, and Changqing Zou. SHREC'14 track: Extended Large Scale Sketch-Based 3D Shape Retrieval. In *3DOR*, 2014. 1, 2, 6, 7, 8

[48] Lei Li, Zhe Huang, Changqing Zou, Chiew-Lan Tai, Rynson W. H. Lau, Hao Zhang, Ping Tan, and Hongbo Fu. Model-Driven Sketch Reconstruction with Structure-Oriented Retrieval. In *SIGGRAPH ASIA*, 2016. 2

[49] Zhigang Li, Gu Wang, and Xiangyang Ji. CDPN: Coordinates-based Disentangled Pose Network for Real-Time RGB-based 6-DoF Object Pose Estimation. In *ICCV*, 2019. 5, 6

[50] Fabian Manhardt, Diego Martı Arroyo, Christian Rupprecht, Benjamin Busam, Nassir Navab, and Federico Tombari. Explaining the Ambiguity of Object Detection and 6D Pose from Visual Data. In *ICCV*, 2019. 5

[51] Daniel Maturana and Sebastian Scherer. VoxNet: A 3D Convolutional Neural Network for Real-Time Object Recognition. In *IROS*, 2015. 3

[52] Umar Riaz Muhammad, Yongxin Yang, Yi-Zhe Song, Tao Xiang, and Timothy M. Hospedales. Learning Deep Sketch Abstraction. In *CVPR*, 2018. 2

[53] Gen Nishida, Ignacio Garcia-Dorado, Daniel G. Aliaga, Bedrich Benes, and Adrien Bousseau. Interactive Sketch-

ing of Urban Procedural Models. *ACM Tans. Graph*, 2016. 3, 7

[54] Luke Olsen, Faramarz F. Samavati, Mario Costa Sousa, and Joaquim A. Jorge. Sketch-based Modeling: A Survey. *Computer & Graphics*, 2009. 2

[55] Kaiyue Pang, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. Cross-Domain Generative Learning for Fine-Grained Sketch-based Image Retrieval. In *BMVC*, 2017. 3

[56] Kaiyue Pang, Yongxin Yang, Timothy M. Hospedales, Tao Xiang, and Yi-Zhe Song. Solving Mixed-Modal Jigsaw Puzzle for Fine-Grained Sketch-based Image Retrieval. In *CVPR*, 2020. 3

[57] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation. In *CVPR*, 2019. 7, 8

[58] Anran Qi, Yulia Gryaditskaya, Jeifei Song, Yongxin Yang, Yonggang Qi, Timothy M. Hospedales, Tao Xiang, and Yi-Zhe Song. Toward Fine-Grained Sketch-Based 3D Shape Retrieval. *TIP*, 2021. 1, 2, 3, 4, 5, 6, 7, 8, 9

[59] Anran Qi, Yi-Zhe Song, and Tao Xiang. Semantic Embedding for Sketch-Based 3D Shape Retrieval. In *BMVC*, 2018. 3, 4

[60] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *CVPR*, 2017. 3, 4, 6

[61] Charles R. Qi, Li Yi, Hao Su, Guibas, and Leonidas J. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In *NeurIPS*, 2017. 3, 6, 7, 8

[62] Jie Qin, Shuaihang Yuan, Jiaxin Chen, Boulbaba Amor, Yi Fang, Nhat Hoang-Xuan, Chi-Bien Chu, Khoi-Nguyen Nguyen-Ngoc, Thien-Tri Cao, Nhat-Khang Ngo, Tuan-Luc Huynh, Hai-Dang Nguyen, Minh-Triet Tran, Haoyang Luo, Jianning Wang, Zheng Zhang, Zihao Xin, Yang Wang, Feng Wang, and Hongyuan Wang. SHREC'22 Track: Sketch-Based 3D Shape Retrieval in the Wild. *Computer & Graphics*, 2022. 3, 6

[63] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *ICML*, 2021. 8, 9

[64] Jose M. Saavedra, Benjamin Bustos, Tobias Schreck, Sang Min Yoon, and Maximiliam Scherer. Sketch-based 3D Model Retrieval using Keyshapes for Global and Local Representation. In *3DOR*, 2012. 2

[65] Aneeshan Sain, Ayan Kumar Bhunia, Pinaki Nath Chowdhury, Subhadeep Koley, Tao Xiang, and Yi-Zhe Song. CLIP for All Things Zero-Shot Sketch-Based Image Retrieval, Fine-Grained or Not. In *CVPR*, 2023. 9

[66] Aneeshan Sain, Ayan Kumar Bhunia, Subhadeep Koley, Pinaki Nath Chowdhury, Soumitri Chattopadhyay, Tao Xiang, and Yi-Zhe Song. Exploiting Unlabelled Photos for Stronger Fine-Grained SBIR. In *CVPR*, 2023. 1

[67] Aneeshan Sain, Ayan Kumar Bhunia, Vaishnav Potlapalli, Pinaki Nath Chowdhury, Tao Xiang, and Yi-Zhe Song. Sketch3T: Test-time Training for Zero-Shot SBIR. In *CVPR*, 2022. 3

[68] Aneeshan Sain, Ayan Kumar Bhunia, Yongxin Yang, Tao Xiang, and Yi-Zhe Song. Cross-Modal Hierarchical Modelling for Fine-Grained Sketch Based Image Retrieval. In *BMVC*, 2020. 5

[69] Aneeshan Sain, Ayan Kumar Bhunia, Yongxin Yang, Tao Xiang, and Yi-Zhe Song. StyleMeUp: Towards Style-Agnostic Sketch-Based Image Retrieval. In *CVPR*, 2021. 2, 3

[70] Patsorn Sangkloy, Nathan Burnell, Cusuh Ham, and James Hays. The Sketchy Database: Learning to Retrieve Badly Drawn Bunnies. *ACM Trans. Graph.*, 2016. 1, 4

[71] Gerald Schweighofer and Axel Pinz. Robust Pose Estimation from a Planaer Target. *IEEE TPAMI*, 2006. 5

[72] Roy Sheffer and Ami Wiesel. PnP-Net: A hybrid Perspective-n-Point Network. In *ECCV*, 2020. 6

[73] Robin Sibson. Information Radius. *Probability Theory and Related Fields*, 1969. 4

[74] K. Simonyan and A. Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv preprint arXiv:1409.1556*, 2014. 3, 4, 6

[75] Jifei Song, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. Fine-grained Image Retrieval: the Text/Sketch Input Dilemma. In *BMVC*, 2017. 3, 4

[76] Jifei Song, Qian Yu, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. Deep Spatial-Semantic Attention for Fine-Grained Sketch-based Image Retrieval. In *ICCV*, 2017. 3

[77] Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik Learned-Miller. Multi-view Convolutional Neural Networks for 3D Shape Recognition. In *ICCV*, 2015. 2, 3, 7, 8

[78] Johan W.H. Tangelder and Remco C. Veltkamp. A Survey of Content Based 3D Shape Retrieval Methods. *MTAP*, 2008. 2

[79] David E. Tyler. Statistical Analysis for the Angular Central Gaussian Distribution on the Sphere. *Biometrika*, 1987. 6

[80] Fang Wang, Le Kang, and Yi Li. Sketch-based 3D Shape Retrieval using Convolutional Neural Networks. In *CVPR*, 2015. 2, 3

[81] Fang Wang, Le Kang, and Yi Li. Sketch-based 3D Shape Retrieval using Convolutional Neural Networks. In *CVPR*, 2015. 7

[82] Jiayun Wang, Jierui Lin, Qian Yu, Runtao Liu, Yubei Chen, and Stella X. Yu. 3D Shape Reconstruction from Free-Hand Sketches. *arXiv preprint arXiv:2006.09694*, 2020. 2, 3, 7, 8

[83] Lingjing Wang, Cheng Qian, Jifei Wang, and Yi Fang. Unsupervised Learning of 3D Model Reconstruction from Hand-Drawn Sketches. In *ACM MM*, 2018. 2

[84] Peng-Shuai Wang, Yang Liu, Yu-Xiao Guo, Chun-Yu Sun, and Xin Tong. O-cnn: Octree-Based Convolutional Neural Networks for 3D Shape Analysis. *ACM Trans. Graph*, 2017. 3

[85] Tong Wu, Liang Pan, Junzhe Zhang, Tai Wang, Ziwei Liu, and Dahua Lin. Density-aware Chamfer Distance as a Comprehensive Metric for Point Cloud Completion. In *NeurIPS*, 2021. 7

[86] Jin Xie, Guoxian Dai, and Yi Zhu, Fan Fang. Learning Barycentric Representations of 3D Shapes for Sketch-based 3D Shape Retrieval. In *CVPR*, 2017. 2, 3, 6

[87] Haoran Xu, Sixing Lu, Zhongkai Sun, Chengyuan Ma, and Edward Guo. VAE based Text Style Transfer with Pivot Words Enhancement Learning. In *ICON*, 2021. 2, 4

[88] Rui Xu, Zongyan Han, Le Hui, Jianjun Qian, and Jin Xie. Domain Disentangled Generative Adversarial Network for Zero-Shot Sketch-Based 3D Shape Retrieval. In *AAAI*, 2022. 3, 6, 7

[89] Gang-Joon Yoon and Sang Min Yoon. Sketch-based 3D Object Recognition from Locally Optimized Sparse Features. *Neurocomputing*, 2017. 2

[90] Sang Min Yoon, Maximilian Scherer, and Tobias Schreck. Sketch-based 3D Model Retrieval using Diffusion Tensor Fields of Suggestive Contours. In *ACM MM*, 2010. 2

[91] Qian Yu, Feng Liu, Yi-Zhe Song, Tao Xiang, Timothy M. Hospedales, and Chen Change Loy. Sketch Me That Shoe. In *CVPR*, 2016. 1, 2, 3, 4, 5, 6, 7, 9

[92] Song-Hai Zhang, Yuan-Chen Guo, and Qing-Wen Gu. Sketch2Model: View-Aware 3D Modeling from Single Free-Hand Sketches. In *CVPR*, 2021. 2, 7, 8

[93] Yue Zhong, Yulia Gryaditskaya, Honggang Zhang, and Yi-Zhe Song. Deep Sketch-Based Modeling: Tips and Tricks. In *3DV*, 2021. 2, 7, 8

[94] Yue Zhong, Yulia Gryaditskaya, Honggang Zhang, and Yi-Zhe Song. A study of deep single sketch-based modeling: View/style invariance, sparsity and latent space disentanglement. *Computer & Graphics*, 2022. 2

[95] Yue Zhong, Yonggang Qi, Yulia Gryaditskaya, Hongang Zhang, and Yi-Zhe Song. Towards Practical Sketch-Based 3D Shape Generation: The Role of Professional Sketches. *IEEE-TCSVT*, 2021. 1

[96] Fan Zhu, Jin Xie, and Yi Fang. Learning Cross-Domain Neural Networks for Sketch-Based 3D Shape Retrieval. In *AAAI*, 2016. 2, 3