

针对浏览式草图造型的按需部件建议技术

Xuekun Guo

2016.04.25

摘要

针对浏览式草图造型的按需部件建议技术

1 引言

开发易用且功能强大的三维造型技术一直是计算机图形学的重要目标之一。专业工具，例如，Autodesk Maya[®]，具有强大的三维造型功能，但操作复杂，需要漫长的培训学习才能掌握。那些易用的工具，例如，Spore Creature Creator[®]将造型过程简化为拼接预定义好的部件。该方法同时提供了简单的部件变形方法，但总体上看，造型能力有限。弥合三维造型工具易用与功能强大之间的鸿沟是图形学界最具挑战性的问题之一。

针对该问题，一些研究人员尝试开发基于草图的造型技术 [LIAL07]。这是一个极具前景的研究方向，因为大多数人都具备基本的绘画技术，都可以使用该类技术造型 [TSH99]。但问题是大多数用户的绘画技术不太好，所以完全按草图很难恢复出具有丰富几何细节的三维模型。最近几年，研究人员尝试通过草图从三维模型数据库内搜索模型[ERB⁺12]，或者通过重新组合现有模型的部件得到新模型[LF08]，实现了新手快速造型高质量模型的功能。然而，由于这类技术仅仅允许组合预定义部件或对部件做简单的变形，限制了造型结果的多样性。可见，它们是典型的易使用但表达能力低的技术。

在本章内，我们提出Sketch2Cut技术，一个基于数据驱动的草图式造型技术。该方法表现力极强，而且保持了易用性。我们的方法不再限定使用预定义的部件组合模型，而是采用实时分割技术，从三维模型数据库内搜索符合造型意图的部件。我们的技术不需要预分割或者标记数据库内的模型。

使用Sketch2Cut，用户可以简单地使用草图创建模型（如图 1）。我们的系统简单易学易用，它除了要求用户具有类似Teddy系统的草图绘制技术外，不需要其它任何造型知识或技术。然而，使用我们的系统又可以创建出更加多样的几何细节丰富的模型。

设计我们的系统面临非常大的技术难点。因为不再有一个预分割（或标记）的模型数据库，我们必须实时地找到并切割下与草图相匹配的部件，这样的搜索空间几乎是无穷的。为了将搜索空间控制在可求解的范围内，我们设计了一种平衡策略。该策略使得所提示的部件在多样化与可用性间取得平衡。此外，我们的匹配方法必须足够快，可以实时地建议候选部件。我们发现如果三维模型的二维投影相似，那该三维模型相似。因此，我们将三维模型间的匹配问题转化为多视角下的二维轮廓匹配问题。

本文贡献有三点：

- 提出一个探索式草图造型方法。该方法从未预分割的三维模型数据库内，实时提取符合用户意图的个性化部件，造型高质量的模型。
- 提出一个快速的基于草图的三维模型局部匹配算法。该算法将三维模型局部匹配问题转化为多视角下的二维轮廓局部匹配。三维模型的二维投影由多个轮廓段（Boundary Contour Section）构成。这些轮廓段组织在一种新的随机混合 k NN图（randomized compound k NN graph, RC- k NNG）内。
- 提出一个新的实时分割算法。该算法基于三维模型的超面片图表示，可以快速地从三维模型上提取与用户草图匹配的三维部件。部件提取采用由粗到细的策略，部件边界逐步优化。

本章接下来的内容安排如下：首先，介绍研究背景及相关工作，包括模型分割问题的基础知识，现有模型语义分割方法与基于草图的模型搜索方法；然后，概述本章方法的流程；接着，介绍快速局部匹配加速结构RC- k NNG；然后，介绍候选模型搜索技术；接着，介绍逐步式部件分割技术；然后，介绍我们技术的应用案例及实验结果；最后，对本章做小结。

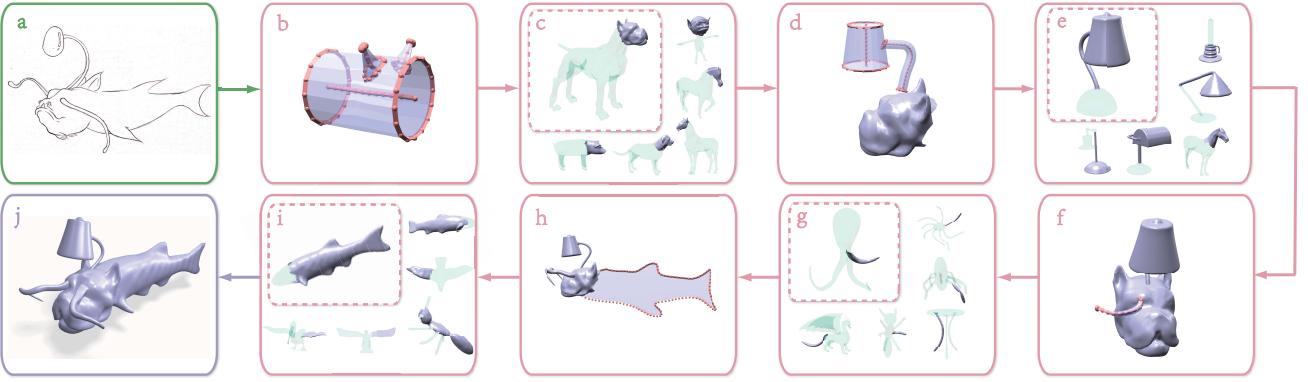


Figure 1: 给一张概念设计图 (a) 做为参考，用户使用我们的方法非常容易地创建具有丰富几何细节的模型 (j)。系统基于草图生成三维代理模型 (b, d, f, h)。代理模型用于实时地从模型数据库内提取个性化的候选部件。我们的方法创建的模型可以由非典型部件（例如，(c) 内头部件与 (e) 内的台灯部件）与来自不同类模型的部件（例如，(c) 内的狗，(e) 内的台灯与 (g) 内的章鱼）构成。

2 相关工作

；

3 方法概述

用户交互 使用我们系统的用户可以先从数据库内选择一个基础模型（某部件或者某模型）。接着，用户通过草图造型代理模型来表示造型意图。系统实时地从数据库内提取一系列与草图相匹配的候选部件。与现有工作显著不同的是，我们提示的部件可以是非常规部件，而现有工作仅可提示预分割的部件（如图 2）。

用户从建议列表内选择满意的部件。该部件自动与当前模型组合到一起。如果没有满意的部件，用户可以改变代理部件重新搜索，得到更多建议部件。该过程不断重复，直至造型出满意的结果。

系统流程 系统流程见图 3。我们的方法由两个阶段构成：离线阶段与在线阶段。离线阶段，我们建立加速结构。该加速结构用于实时地提取用户所需部件。在线阶段，设计人员与系统交互创建模型。

离线阶段 首先，我们为数据库模型提取不同视角下的边界轮廓 [DFRS03]。接着，在不同尺度下提取边界轮廓描述符。

然后，我们将所有模型的边界轮廓组织到RC- k NNG内，用于部件快速搜索（见第 4节）。

最后，我们为数据库模型建立超面片图表示（见第 6.1节）。超面片指模型表面上一组相邻面片。将模型分解成一系列超面片，是为了得到该模型的低复杂度表示方法。三维模型的超面片表示使得我们可以快速提取与匹配的三维部件。

在线阶段 通过比较代理轮廓与模型轮廓，我们将三维局部匹配问题转化为多视角的二维轮廓匹配问题，从而找到候选部件包含的超面片。用户选择候选部件，我们将该部件从其模型上切割下来并与当前模型组合到一起。我们使用由粗到细的模型分割策略：先在超面片层次计算粗糙边界，再在面片层次优化该边界。与其它分割方法[Sha08]不同，我们的分割方法将轮廓感知，模型表面凹凸性，模型表面光滑程度考虑进来（见第 6节）。

4 快速局部匹配结构

在运行阶段，用户绘制代理模型轮廓，表达建模意图。系统快速从数据库内搜索出与代理模型相匹配的部件。我们的系统可以搜索出模型上任意部件，而不仅仅是预分割出的标准部件。因此，我们必须实时地处理数据库内每个模型，找到模型轮廓上与代理模型轮廓相匹配的部分。这要求一个极其快速的局部匹配方法。

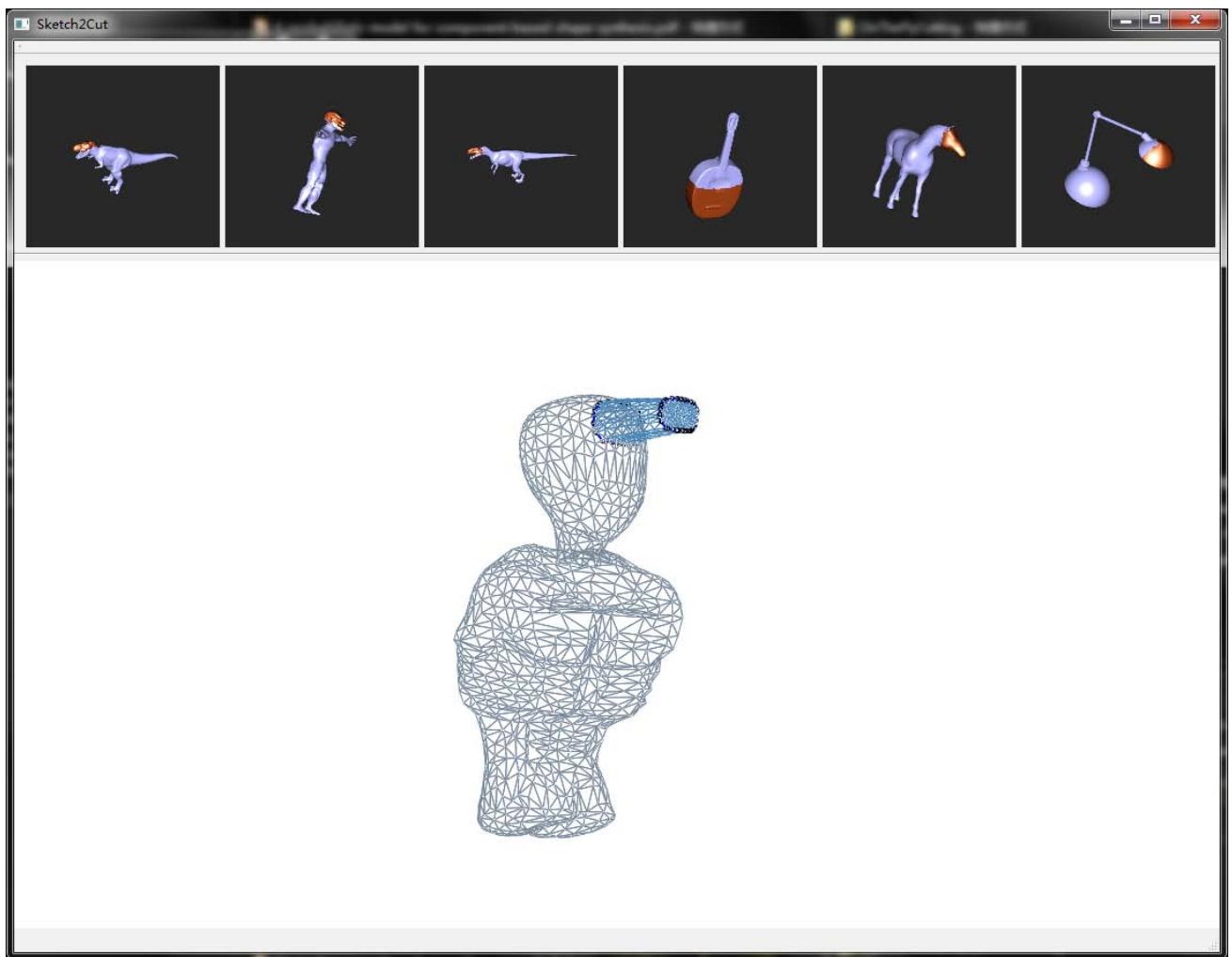


Figure 2: 交互界面截图。界面内展示了当前模型，用户草图，代理模型，系统建议的非常规部件。

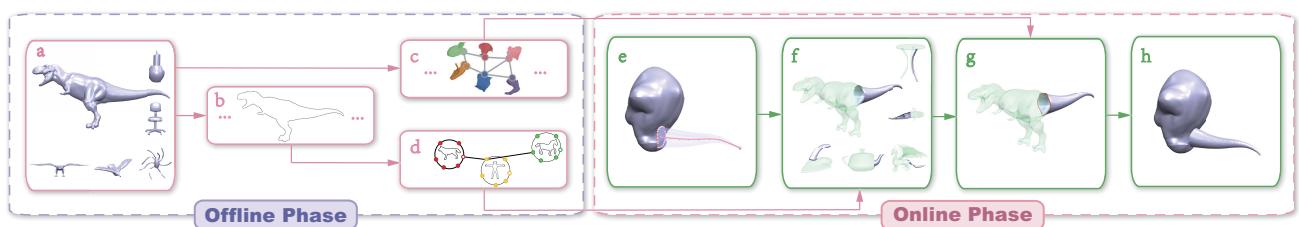


Figure 3: 系统流程示意图。离线阶段，我们首先为数据库模型（a）提取边界轮廓（b）；然后，建立三维模型超面片图（c）；最后，将所有模型的所有轮廓线组织到RC-KNNG内（d）。在线阶段，用户首先通过草图（红色）创建代理模型（当前模型内浅蓝色部件）表达造型意图（e）。系统通过模型局部匹配技术实时地提取与草图相匹配的部件（f），并按顺序提示给用户（浅蓝色）。用户选择满意的部件后，系统将它从三维模型上切割下来（g）并组合至当前模型（h）。

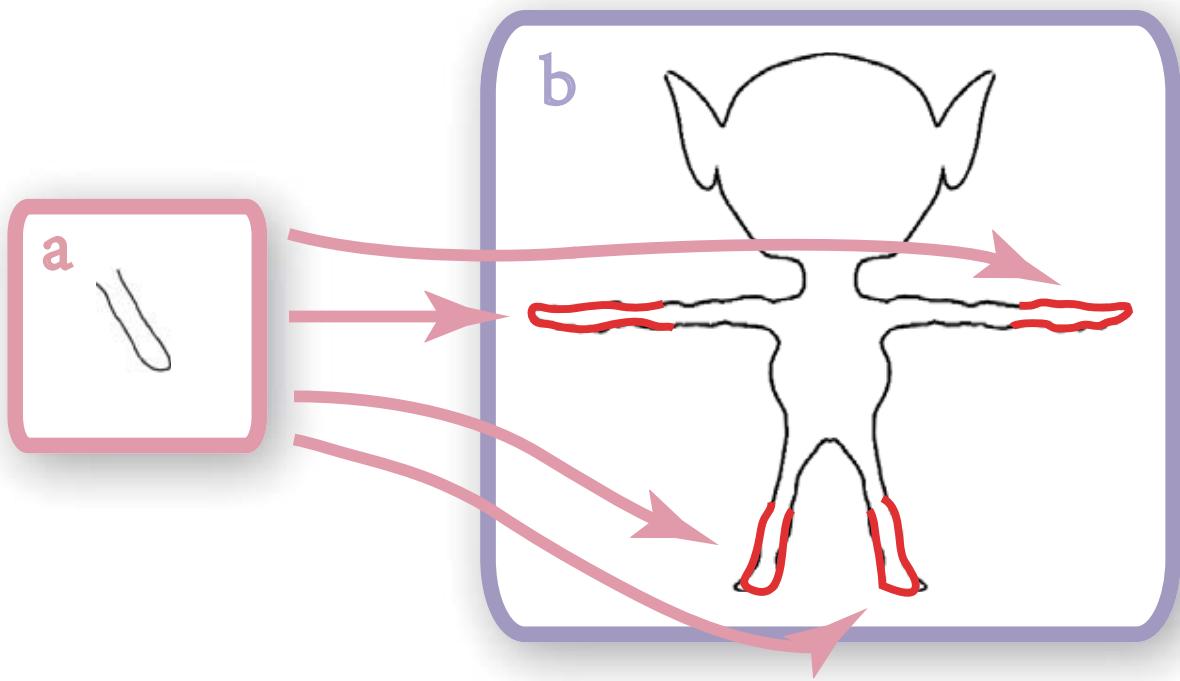


Figure 4: 查询轮廓与模型轮廓的不同部分相匹配。查询轮廓 (a) 在四个不同的段上与模型轮廓 (b) 相匹配 (标为红色)。

如果两个三维模型相似，那么他们从所有角度上看起来相似 [CTSO03]。我们发现该观察结论同样适用于代理模型与模型上的目标部件。因此，为了找到数据库模型上与代理模型相匹配的部分，我们可以比较代理模型与数据库模型间的二维轮廓。如果我们使用用户草图与数据库模型轮廓做局部匹配，一些与用户草图相似的部件将会被丢掉。如果我们直接将代理模型与数据库模型做局部匹配，这非常耗时，无法满足交互需求。因此，我们将代理模型轮廓与数据库模型在多个视角下的投影轮廓做局部匹配。在离线阶段，我们求得每一个数据库模型在很多视角下的边界轮廓。然后，问题转化为将代理模型轮廓与数据库模型轮廓做局部匹配的问题。

实时地将如此多的数据库模型轮廓与代理模型轮廓做局部匹配异常困难。现有的算法，比如，分割树 [ML14]、哈希存储 [AI08]、KNN图 (k -nearest neighbors graph) [WWZ⁺12]等，都针对全局匹配，而不是我们这里的局部匹配。

为了快速找到数据库模型轮廓线与代理模型轮廓线相匹配的部分，我们提出一种新的数据结构，随机混合KNN图 (RC-KNNG)。RC-KNNG的一个顶点代表数据库模型的一条边界轮廓线。然而，在我们的局部匹配问题中，单个轮廓可能在多个不同的段与查询轮廓相匹配 (如图 4)。为了反映这一点，RC- k NNNG允许一个顶点与多组 k 个最近顶点相连。与当前顶点相连的每一组顶点都与当前轮廓线上同一段相匹配。因此，通过与一条轮廓线的局部匹配结果，我们可以快速得到在其它轮廓线上的局部匹配结果。

创建RC- k NNNG 通过暴力方法计算模型轮廓上所有相似的轮廓段建立RC- k NNNG过于耗时，不可行。因此，我们采用双随机策略建立RC- k NNNG：

1. 我们首先为每条模型轮廓线在每个尺度下提取 n (在我们的实验中， $n = 6$) 段 (如图5 (a))。一个尺度由一组在模型轮廓上均匀分布的采样点构成：大尺度的采样间距小，包括更多采样点。我们采用3个尺度，即每条模型轮廓线上的采样点数分别为50, 150, 250。
为了得到轮廓线上一段，我们随机选择一个不在其它轮廓段上的采样点。选择该采样点的概率是 p ($p \propto c \cdot d$, c 为该采样点的曲率, d 为该采样点与最近的在其它轮廓段上的采样点的距离)。以该采样点为中心，分别向两侧取10个采样点，构成轮廓段。
2. 我们采用多重随机分治策略 [WWZ⁺12]为所有的轮廓段建立KNN图 (如图5 (b))。

我们将所有的有效轮廓段添加到全局有效轮廓段列表内。有效轮廓段指该轮廓段与它最近的 k 个邻居间距 (见5.2节) 低于阈值 $\epsilon = 0.85$ 。然后，我们对全局有效轮廓段列表内的轮廓段做聚类，聚类代表是聚类中心轮廓段。这些代表做为种子轮廓用子查询阶段 (如图5 (c))。聚类个数为 $N/80$ (N 是全局有效轮廓段列表内轮廓段的个数)。

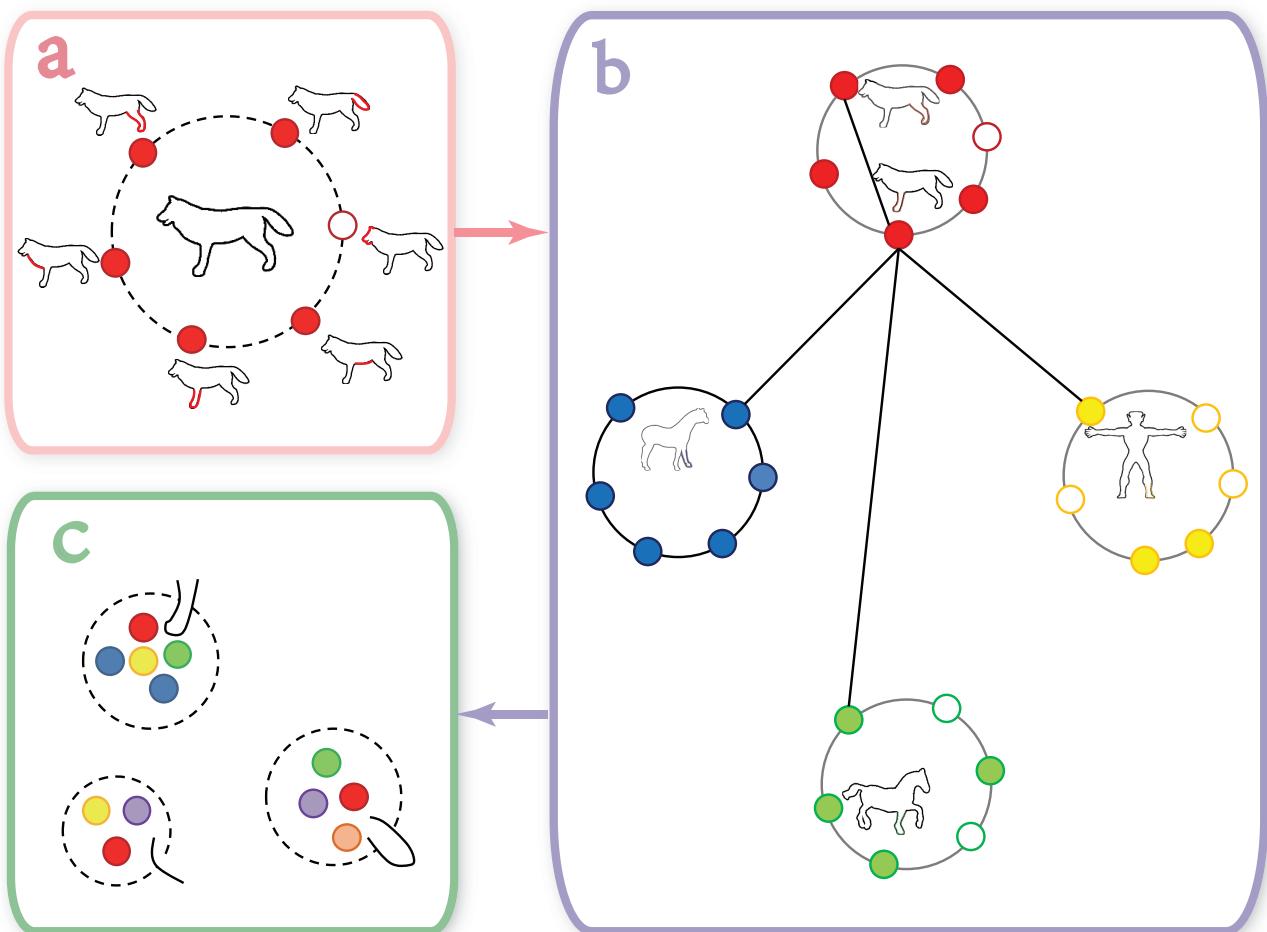


Figure 5: 随机混合KNN图。实线圆表示有效轮廓段，虚线圆表示无效轮廓段。给一条模型轮廓（做为随机混合KNN图的一个顶点），我们首先取得若干轮廓段（a）。然后，为每一条轮廓段找到最近邻居并在各轮廓段所属的轮廓间添加边（b）。最后，找到有效轮廓段并做聚类（c）。

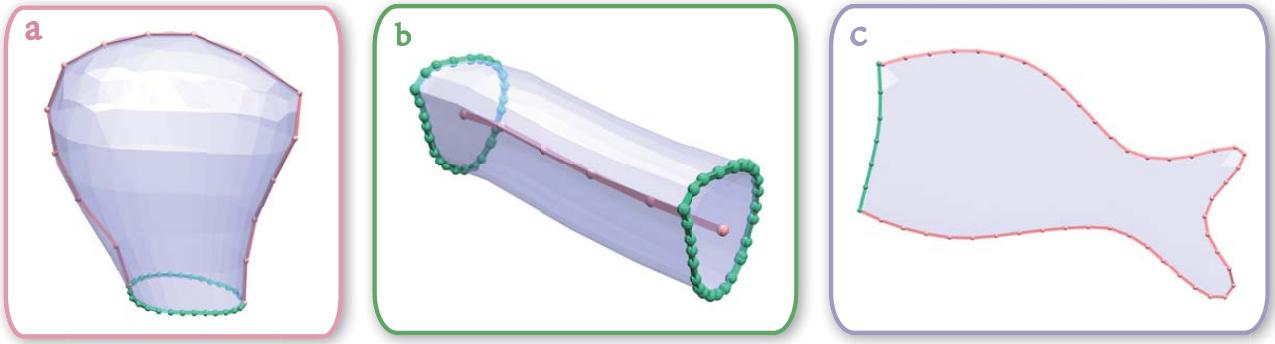


Figure 6: 三种基础代理模型。 (a) 广义球体。 (b) 广义柱体。 (c) 薄板模型。

5 候选模型搜索

本章中，我们介绍RC- k NNG如何从未做预分割的模型数据库内快速地精确地搜索出与代理模型相匹配候选部件。

5.1 代理模型创建

与 [FKS⁺04]类似，我们使用三维代理模型从数据库内搜索部件并辅助部件组合。我们提供三种基本代理模型，从而使用户更容易表达造型意图。当然，用户输入的草图也可直接用于从数据库内搜索部件。

我们提供的三种基本代理模型是：广义球体，广义柱体，薄板模型（如图6）。广义球体通过Nealen等人 [ATOM07]的方法生成。广义柱体通过两笔生成：在当前模型上画一个圆环表示柱体底座，然后用一条曲线指示中心线。中心线位于用户视线方向与圆球起点定义的空间平面上。广义柱体的一端可以放缩，得到圆锥或圆台。要得生成薄板型代理模型，用户需要首先在当前模型上画一条曲线，再绘制出其轮廓。该轮廓位于用户视线方向与曲线起点定义的空间平面上。我们的系统允许用户对代理模型施加各种操作，包括放缩，切割，拖拽顶点等。通过如上方法生成的各种代理模型见图6及本文视频。

用户创建代理模型过程中，系统记录每次操作的视角及操作前后代理模型的体积变化。代理模型建成后，我们提取代理模型在体积变化最大的视角下的二维轮廓，用于描述用户的建模意图（期望的部件）。这里的体积变化指画草图前后代理模型的体积之差。

5.2 轮廓描述符

查询轮廓（代理模型轮廓）整体做一个轮廓段。模型轮廓在每个尺度下都由若干轮廓段构成（见4节）。模型轮廓的轮廓段与查询轮廓相匹配，从而得到查询轮廓与模型轮廓的局部匹配。

为了比较两个轮廓段，我们为它们计算轮廓描述符。我们采用Riemenschneider等人 [RDB10]提出的角度矩阵做为轮廓描述符。该描述符描述了轮廓段上采样点间的空间分布。

一个轮廓段内有 $m = 21$ 个采样点。给一对采样点 (b_i, b_j) ，我们计算角度值 α_{ij} ：

$$\alpha_{ij} = \begin{cases} \frac{\langle \overrightarrow{b_i b_j}, \overrightarrow{b_i b_{i+\Delta}} \rangle}{\langle \overrightarrow{b_i b_j}, \overrightarrow{b_i b_{i-\Delta}} \rangle} & \text{if } i < j, \\ \frac{\langle \overrightarrow{b_i b_j}, \overrightarrow{b_i b_{i-\Delta}} \rangle}{\langle \overrightarrow{b_i b_j}, \overrightarrow{b_i b_{i+\Delta}} \rangle} & \text{if } i > j, \\ 0 & \text{if } \|i - j\| \leq \Delta \end{cases}$$

其中， $\delta = 2$ 是偏移量， $\langle l_1, l_2 \rangle$ 代表直线 l_1 与 l_2 间的夹角。轮廓描述符即为该角度值构成的矩阵 A ($A_{ij} = \alpha_{ij}$)。两个轮廓段 S 与 T 间的距离是两轮廓段的角度矩阵 A^S 与 A^T 的欧式距离：

$$D(S, T) = \frac{1}{m^2} \sum_{i=1}^m \sum_{j=1}^m (A_{ij}^S - A_{ij}^T)^2$$

5.3 模型搜索

我们需要先找到 $c = 9$ 个与代理模型匹配上的数据库模型（称为候选模型），然后再从候选模型中提取出候选部件建议给用户。为此，我们在RC- k NNG内找到很多（ $21c$ ）匹配轮廓段，得到对应的匹配模型并为它们计算排序分数。得分最高的 c 个模型会显示给用户。

为得到这些匹配的轮廓段，我们首先将查询轮廓段与RC- k NNG的所有种子轮廓段比较。比较结果存储至优先队列中，与查询轮廓段间距最小的轮廓段排在队首。然后，以最好优先方式（Best-first Search）遍历整个RC- k NNG。遍历过程中，首先优先队列队首出队并加入匹配轮廓段集合，该队首的所有邻居与查询轮廓段做匹配并入队。该过程一直进行，直至匹配轮廓段集合内元素数目为 $21c$ 。

对数据库模型，我们按如下方法计算其做为候选模型的分数：

$$s = \frac{\alpha}{t} \sum_{i=1}^t D_i + \frac{\beta}{t},$$

其中， t 是属于该模型的匹配轮廓段数目， D_i 是第*i*个匹配轮廓段与查询轮廓段间距， $\alpha = 0.95$ ， $\beta = 0.05$ 。

6 逐步式部件提取

用户选择了候选模型，我们必须实时地精确地找到并切割下匹配的候选部件。要实现该目标，我们需要解决如下技术难点：

- 一个轮廓段可能包括多个语义部件。语义部件指类似头部件，椅子背部件等部件。
- 轮廓边界可能是非常规则的或并不与语义部件直接相关。因此，这样的部件仅能以用户草图为参考分割。
- 分割速度要足够快，满足交互需求。

前两种情况如图7所示。

我们通过为数据库模型建立超面片图（Superface Graph, SFG）表示解决如上问题。超面片图是三维模型的离散细粒度表示形式。图上顶点代表超面片。超面片通过对模型做基于边界特征的过分割得到。图上边描述超面片间的相邻关系。边的权重表示相连的超面片组成同一部件的概率（如图8所示）。

超面片图包含了相邻超面片合并为更大超面片的先验概率。我们在离线阶段得到每个数据库模型的超面片图表示。

6.1 超面片图表示

为了给数据库模型建立超面片图表示，我们将模型表面分割成50小块 [HKG11]。一小块视为一个超面片（对应超面片图内一个顶点）。每一对相邻超面片通过超面片图上一条边相连。

为了计算边的权重，我们对三维模型做900次Randomized Segmentation [GF08]。分割的目标部件数目由2变化到10。给一个分割结果，我们为每一个超面片建立一个直方图，描述该超面片属于各分割部件的概率。直方图的一个直方条针对一个分割部件，其值是该超面片位于该分割部件内的面积比例（如图9）。数学上讲，超面片 s 针对分割部件 g 的直方条 h_s^g 定义如下：

$$h_s^g = \frac{\sum_{f \in Faces(g) \cap Faces(s)} Area(f)}{\sum_{f \in Faces(s)} Area(f)},$$

其中， $Face(x)$ 是属于 x 的面片集合， $Area(f)$ 指面片 f 的面积。

这些直方图做为计算超面片位于同一部件内概率的特征向量。我们定义 $P(s, s')$ 为相邻超面片 s 与 s' 位于同一部件上的概率（称为超面片间一致性）。 $P(s, s')$ 为两超面片对应的直方图 H 与 H' 的 χ^2 距离：

$$P(s, s') = 1 - \chi^2(H, H') = 1 - \frac{1}{2} \sum_{g_i \in G} \frac{(h_s^{g_i} - h_{s'}^{g_i})^2}{(h_s^{g_i} + h_{s'}^{g_i})^2},$$

其中， $G = \{g_i | 1 \leq i \leq N\}$ 是一次Randomized Segmentation得到的部件的集合， g_i 是第*i*个分割部件， N 是分割部件个数。

为了得到超面片图上边的权重，我们首先将两超面片在不同Randomized Segmentation结果上的一致性累加，然后通过分割次数归一化。

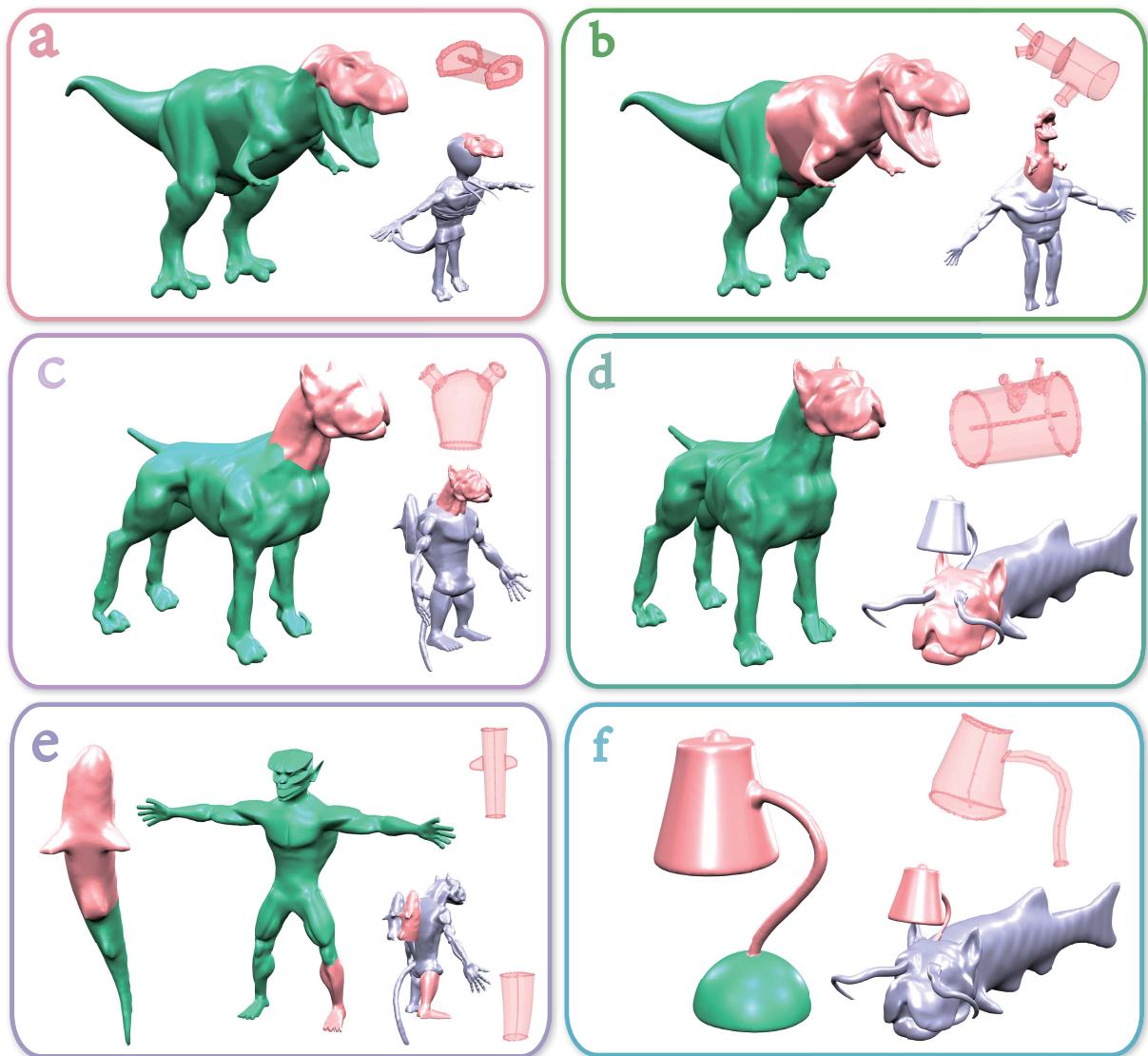


Figure 7: 非常规分割方式示意图。在每个子图内，左侧是数据库模型的分割结果，右下角给出结果模型。

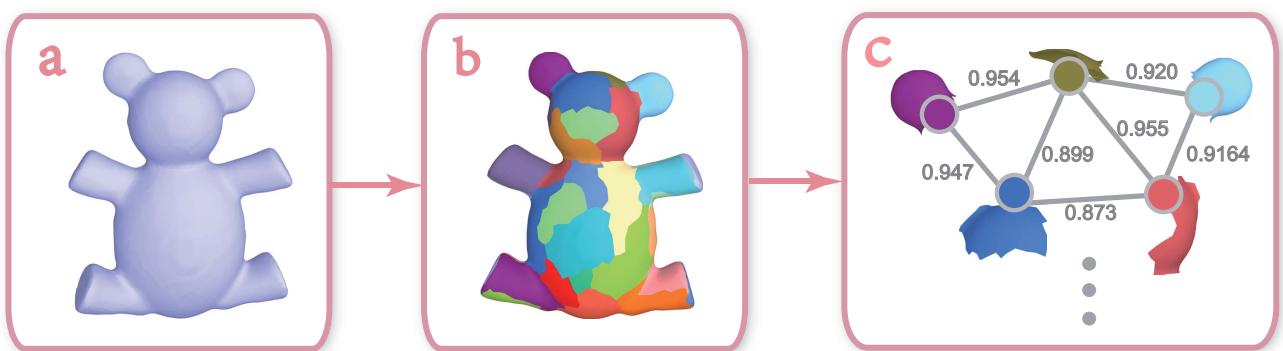
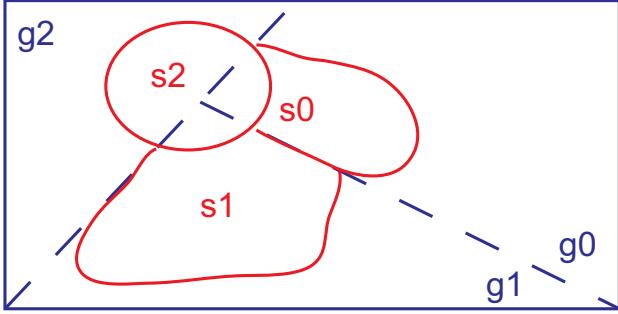


Figure 8: 三维模型超面片图建立过程。给一个Teddy模型（a），我们将它分割成50个超面片（b）。一个超面片是超面片图的一个顶点。相邻超面片由带权重的边连接（c）。



	g_0	g_1	g_2
$Histogram(s_0)$	= 1.0	0.0	0.0
$Histogram(s_1)$	= 0.0	1.0	0.0
$Histogram(s_2)$	= 0.2	0.15	0.65

Figure 9: 超面片在分割部件上的分布示意图。 s_0 , s_1 和 s_2 是超面片。 g_0 , g_1 与 g_2 是在模型上通过Randomized Segmentation得到的分割部件。

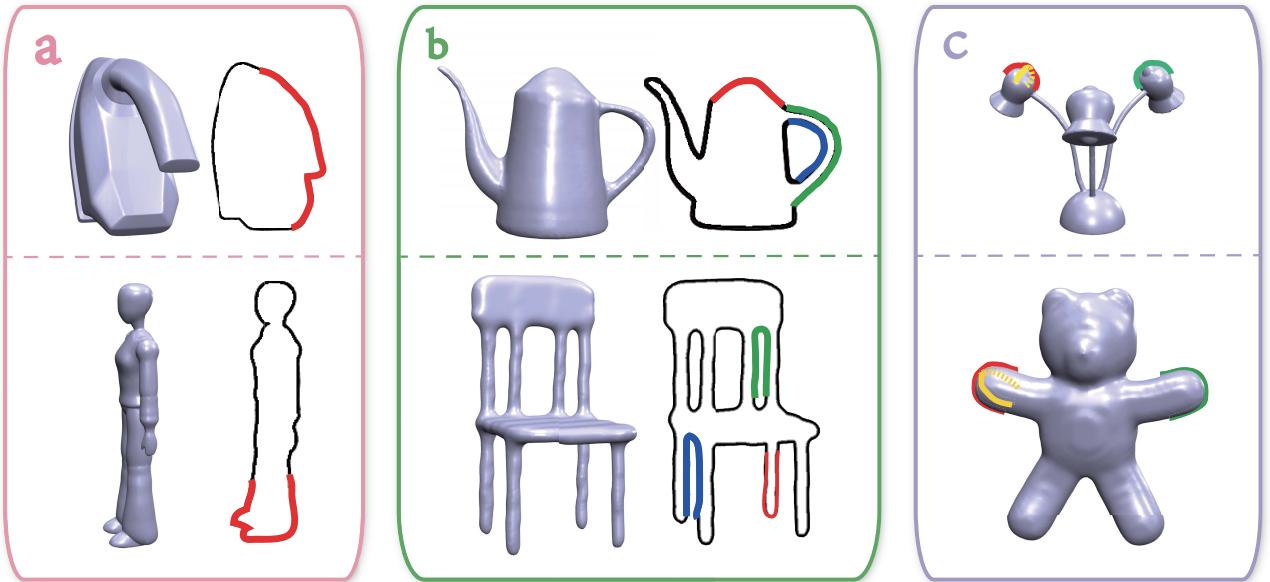


Figure 10: 复杂轮廓及其关系示意图。 (a) 示意图深度不连续轮廓段。 (b) 示意图空轮廓段。 (c) 示意图匹配至同一三维部件在不同视角下的轮廓段。

6.2 模糊部件识别

第一步，我们在模型的超面片图上提取一个与轮廓段大致匹配的三维部件。我们将轮廓段两端点相连得到封闭轮廓段。这形成了用户期望部件的二维投影边界。然后，我们从与投影区域中间最近的超面片图开始，采用泛洪算法，将所有处于该投影区域内的超面片收集起来。

在这个步骤中，我们必须处理如下问题：（1）轮廓段可能深度不连续，跨越几个不相连的部件（10 (a)）。（2）匹配上的轮廓段可能不包围任何部件（图10 (b) 内蓝色与绿色轮廓段包围的部分）。（3）查询轮廓可能与同一三维部件在不同视角下的轮廓段匹配上（如图10 (c)）。为了解决该问题，我们过滤掉无效部件并去掉重复部件。

6.3 由粗到细地边界优化

粗糙部件被识别出来后，我们就把它从模型上分割下来并优化其边界使与代理模型更加匹配。在该过程中，我们考虑如下因素：

- **轮廓闭包** 正如前面所指出的，连接轮廓段两端点的直线与轮廓段构成目标三维部件的自然边界。因此，目标三维部件的投影应该与该边界尽量一致。
- **超面片同现** 如果两个超面片总是出现在随机分割得到的同一部件上，那么它们倾向于同时属于或同时不属于目标三维部件。这一点是超面片的先验概率。

- **凹陷** 模型凹陷信息对于得到高质量的分割结果至关重要，因为凹折痕通常被人类视为自然的分割边界[AZC⁺12]。
- **光滑** 分割部件的边界最好不要是剧烈的锯齿形。

6.4 粗层次部件提取

我们将粗层次部件提取问题形式化为二值标记问题，该问题可通过最小化Gibbs能量求解：

$$E_c = \sum_{i \in V} E_1(l_i) + \sum_{(i,j) \in E} E_2(l_i, l_j),$$

其中， V 与 E 分别代表超面片图 \mathcal{G} 的顶点集合与边集合。 $E_1(l_i)$ 是将顶点*i*标记为 l_i 时的能量耗费。 $E_2(l_i, l_j)$ 是当分别将顶点*i*与*j*标记 l_i 与 l_j 时的能量耗费。 E_l 定义如下：

$$E_1(l_i) = \begin{cases} -\ln \mathcal{P}(i), & l_i = 1, \\ -\ln(1 - \mathcal{P}(i)), & l_i = 0, \end{cases}$$

其中， $\mathcal{P}(i)$ 是将标记1置给顶点*i*的概率。根据轮廓闭包约束， $\mathcal{P}(i)$ 定义为第*i*个超面片的投影位于该轮廓内的面积百分比：

$$\mathcal{P}(i) = \frac{\text{Area}_{proj}^{in}(i)}{\text{Area}_{proj}(i)},$$

其中， $\text{Area}_{proj}^{in}(i)$ 是第*i*个超面片的投影与轮廓线闭包相交部分的面积。 $\text{Area}_{proj}(i)$ 是第*i*个超面片的投影的面积。 E_2 基于超面片同现约束定义为：

$$E_2(l_i, l_j) = \begin{cases} 0, & l_i = l_j, \\ e_{ij}, & l_i \neq l_j, \end{cases}$$

其中， e_{ij} 是超面片图的边权重。该二值标记问题通过二值图割方法求解[BK04]。

6.5 细层次部件提取

在超面片层次上得到目标三维部件（一组相邻的超面片）后，我们将边界超面片投影至原模型上与该超面片中心最近的顶点上。对于超面片图上的边 e ，令 v_1 与 v_2 分别为 e 相连的两超面片在原模型上对应的投影顶点。我们沿着直线 v_1v_2 ，将 v_1 传播至 v_2 。

然后，我们通过迭代地最小化如下能量函数实现边界优化：

$$E_f = E_v + E_s,$$

其中， E_v 是凹陷能量项， E_s 是光滑能量项。 E_v 定义为所有边界边的凹陷能量之和[KT03]：

$$E_v = \sum_{e \in \partial Q} \eta(1 + \cos \alpha_e)|e|,$$

其中， e 是候选三维部件 Q 边界上的一条边， $|e|$ 表示 e 的长度， α_e 是 e 的二面角。 e 是凹陷边时， $\eta = 0.1$ ；否则 $\eta = 1.0$ 。 E_s 定义如下：

$$E_s = \sum_{v \in \partial Q} |\sin \langle e_l, e_r \rangle|,$$

其中， v 是边界 Q 上的顶点。 e_l 是 Q 上指向 v 的一条边， e_r 是 Q 上起于 v 的一条边。我们通过在边界顶点上施加snake operations [JLCW06]最小化能量函数 E_f 。

用户点击鼠标即可执行部件边界优化。该步骤是可选的，它并不会对最终造型结果有太大影响。

7 实验结果

我们用C++语言实现系统原型，并在标准台式机（3.5 GHz Intel Core I7-3770K processor, 8GB memory, and an Nvidia GeForce GTX 660 GPU video card）上做实验。图1给出一个例子的造型流程。给一个概念设计，用户通过草图表达其造型意图。系统基于草图生成三维代理模型并实时地从数据库内提取建议部件。这些建议部件有非典型部件（（c）内的头部部件），由一组常规部件构成的部件（（e）内的台灯部件）及来自不同类模型的部件（例如，（c）内来自狗模型的部件，（g）内来自章鱼模型的部件）。图11给出更多的造型结果。可见，用户可以使用我们的技术生成很多极具创意的结果。

7.1 应用

我们提出的实时的基于草图的分割方法是一种通用性很强的技术，有很多潜在应用。这里我们给出一些应用的例子。用我们方法造型出的模型可以制造成艺术品，例如个性化的设计¹²。

模型变异 本章提出的方法可用于变异模型，如图13所示。给一个分割后的模型，将各部件做为查询模型去数据库内搜索相似部件，从搜索结果中选择一个替换原部件即可得到变异模型。为了得到查询部件对应的轮廓，我们首先对查询部件做主成分分析，然后，将查询部件沿第三主成分方向投影得到对应轮廓。

部件建议 我们的实时部件提取技术可用于提示可用部件 [CK10, CKGK11]。与以前的方法不同，我们不需要预分割的三维模型数据库。给一个查询模型，其轮廓做为我们技术的输入，在数据库模型上得到匹配的部分，然后将数据库模型上与匹配部分互补的部分提示给用户。提示的部件可以是非常规则部件或来自不同类的部件（如图14）。

多尺度部件建议 我们的系统可建议不同尺度的部件。这里的“尺度”指所建议部件的大小。给一个按如上方法提示的部件，我们在该部件对应的超面片图上做Normalized Cuts [GF08]，得到 T_n 个分割片段，每一个分割片段都由一组相邻超面片组成。 T_n 定义如下：

$$T_n = \frac{w(Vol(P_M) - Vol(p_p))}{Vol(p_p)},$$

其中， $Vol(\cdot)$ 是模型的体积， P_M 是搜索得到的模型， p_p 是数据库模型上与查询部件相匹配的部件， $w = 2, 1 \leq T_n \leq 7$ 。我们将分割片段按与匹配部件的距离由近及远排序。用户选择一个尺度 S ，我们将小于等于该尺度的分割片段做为建议部件提示给用户（如图15）。

7.2 快速的轮廓感知的局部匹配

我们的数据库内共有513个三维模型。这些模型共提取出10,773个轮廓。将这些轮廓组织成RC-*k*NNG约用3.5小时。建立一个模型的超面片图表示平均用时20秒。基于超面片图表示，可实时提取三维目标部件。粗层次部件提取与细层次部件提取分别平均用时0.15与10.3毫秒。基于两个加速结构（RC-*k*NNG与SFG）与由粗到细的部件提取策略，我们的系统可以实时响应用户操作。用户输入草图后，平均用0.56秒（通过GPU与多核加速策略）提示可用部件。

为了评估我们的RC-*k*NNG结构，我们与如下方法做了对比：

- 通过暴力方法建立的传统*k*NNNG（单层）（简称*k*NNNG）
- 通过Wang [WWZ⁺12]提出的随机近似*k*NNNG方法建立的传统*k*NNNG（简称Wang's *k*NNNG）
- 通过暴力方法建立的RC-*k*NNNG
- 通过Wang提出的随机近似*k*NNNG方法建立的RC-*k*NNNG

对两个*k*NNNG方法（1与2），图的顶点表示一整条模型轮廓线。边及其权重通过轮廓线全局匹配得到。在与代理轮廓线做局部匹配时，我们必须显示地从模型轮廓上提取轮廓段，这极大地降低了方法1与2的速度。

在我们的实验中， $k = 20$ ，每个模型轮廓线上提取6条轮廓段。

四种方法的搜索结果见表1。可以很明显地看出，本文提出的方法效果最好。我们的方法仅比Wang's *k*NNNG多花2倍时间用于建立加速结构。但从搜索时间上来看，我们的方法是Wang's *k*NNNG的97倍。*k*NNNG方法（方法1与2），需要实时做局部匹配，所以要慢得多。

7.3 对比

为了定性与定量地评估我们方法的效果，我们对比了如方法：

- **Ours.** 本文提出的方法

Table 1: 四种不同方法的搜索效果对比。R-time表示搜索时间（以秒为单位）。C-time表示创建对应数据结构所用时间（以小时为单位）。

Algorithm	Performance	R-time	C-time
Brute force k NN	4.54	51.7	
Wang's k NN approximation method	5.68	1.15	
RC- k NN with the brute force method	0.053	33.89	
RC- k NN with Wang's method	0.058	3.52	

Table 2: 非典型部件与典型部件在模型内所占比例统计结果。“Ours::Atypical”与“Ours::Regular”表示Ours系统造型结果内非典型部件与典型部件的比例。“PreSeg::Atypical”与“PreSeg::Regular”分别表示PreSeg系统造型结果内非典型部件与典型部件的比例。

Type	STAT.	Ave.	SD.	Max.	Med.	Min.
Ours::Atypical	0.26	0.24	0.67	0.25	0.0	
Ours::Regular	0.74	0.24	1.0	0.75	0.33	
PreSeg::Atypical	0.0	0.0	0.0	0.0	0.0	
PreSeg::Regular	1.0	0.0	1.0	1.0	1.0	

- **PreSeg.** 基于本文方法实现的基于预分割的模型数据库的方法。该方法实现细节如下：1) 每个数据库模型都预分割成常规部件（典型语义部件）。比如，人模型分成4部分：头部，躯干，胳膊，腿。2) 为每一个部件提取各视角下的边界轮廓线。3) 将所有部件的轮廓线组织在 k NN图内。图内顶点是部件轮廓。通过部件轮廓全局匹配为顶点建立边。3) 在运行阶段，输入三维代理部件，系统采用类似Ours的方法通过全局匹配返回一组候选部件。

我们邀请4位三维造型艺术家与16位学生参与用户测试。4位艺术家中2位具有6年以上三维造型经验，另外2位有4年经验。所有的参与者位于22到31岁之间。我们将参与者分成A, B两组。每一组都由2位艺术家与8位学生构成。在每一组内男女人数相同。我们设计了两个开放式造型任务：

- **生物造型** 假设你是Valve®公司的一名三维造型师。项目组计划为电子游戏Dota 2®加入一个新英雄。请使用我们的造型系统设计该角色的原型。
- **日常用品造型** 假设你是创意生活公司的一名设计师。要求你为客户设计一个个性化的日常用品（例如，个性化的台灯，创意服饰等）。请使用我们的系统设计该日常用品的原型。

用户测试第一阶段 第一阶段的目标是验证非典型部件（非常规部件，由一组常规部件构成的部件）对用户造型是否有用。我们让所有的测试者完成两个造型任务。A组用户首先使用Ours系统，然后是PreSeg系统。B组用户首先使用PreSeg系统，然后是Ours系统。在用户使用系统之前，我们先用20分钟时间向其介绍系统操作方法。之后，用户花30分钟操作系统，包括三维代理部件造型，浏览建议部件与组合部件。图16展示了一位测试者正在做用户测试。该阶段的一部分造型结果见图17。我们统计了非典型部件与典型部件在各模型内的比例（表2）。很明显，Ours系统的造型结果可以由非典型部件构成。然而，PreSeg系统的造型结果仅由典型部件构成。Ours系统提供的非典型部件有利于用户的造型。

用户测试第二阶段 在第二阶段，我们的目标是验证Ours系统建议的部件是否比PreSeg系统建议的更能激发造型灵感。A组测试者先使用PreSeg系统，然后是Ours系统。B组用户相反。我们要求所有测试者为系统提供的所有建议打分。分数成为5等：5分表示非常具有启发性，1分表示非常平庸。

表3给出了Ours系统与PreSeg系统建议的部件的得分情况。由表我们可以很明显地发现Ours系统建议部件的得分高于PreSeg系统的。在图18内，我们给出一组代表性的建议部件及其得分。

为了定量地评估两个系统建议部件的非规则性，我们设计了非规则度量标准。直观地讲，不规则度量描述了当前部件与同一模型上相关典型部件间的差异程度。给一个部件 p (p 来自三维模型 M)，其非规则程度值 $IRR(p)$ 定义如下：

$$IRR(p) = \max \{d(p, q_i) | \{q_i\} = r(p)\},$$

其中， $r(p)$ 是 M 上与 p 相交的典型部件的集合。 $d(p, q_i)$ 是 p 与 q_i 的Light Field Descriptor [CTSO03]之差的归一化值。

Table 3: 建议部件得分统计结果。“Ours::Total”与“PreSeg::Total”分别表示Ours系统与PreSeg系统建议部件的得分。“Ours::Atypical”与“Ours::Regular”分别表示Ours系统建议的非典型部件与典型部件的得分。“PreSeg::Atypical”与“PreSeg::Regular”分别表示PreSeg系统建议的非典型部件与典型部件的得分。

Type \ STAT.	Ave.	SD.	Max.	Med.	Min.
Ours::Total	3.59	1.03	5	3	1
PreSeg::Total	3.06	0.74	5	3	1
Ours::Atypical	4.47	0.72	5	5	2
Ours::Regular	3.02	0.78	5	3	1
PreSeg::Atypical	-	-	-	-	-
PreSeg::Regular	3.06	0.74	5	3	1

Table 4: 非规则度量值统计结果。“Ours::Total”与“PreSeg::Total”分别代表Ours系统与PreSeg系统建议部件的非规则度量值。“Ours::Atypical”与“Ours::Regular”分别代表由Ours系统建议的非典型部件与典型部件的非规则度量值。“PreSeg::Atypical”与“PreSeg::Regular”分别代表PreSeg系统建议的非典型部件与典型部件的非规则度量值。

Type \ STAT.	Ave.	SD.	Max.	Med.	Min.
Ours::Total	0.024	0.033	0.098	0	0
PreSeg::Total	0	0	0	0	0
Ours::Atypical	0.062	0.022	0.098	0.053	0.025
Ours::Regular	0	0	0	0	0
PreSeg::Atypical	-	-	-	-	-
PreSeg::Regular	0	0	0	0	0

表4给出了两个造型系统建议部件的非规则度量值。由表可以很明显地看出，Ours系统建议的部件相对PreSeg系统的更加不规则。非典型部件导致了非规则性。图19给出了一组代表性的建议部件及其非规则度量值。

我们统计了Ours系统与PreSeg系统提供的非典型部件与典型部件的得分（表3）。可以很明显地发现，非典型部件的平均得分高于典型部件的。表5给出了每一个分值内非典型部件与典型部件所占比例。很显然，非典型部件得分高于典型部件。由此，可以得出结论Ours系统建议的部件相对PreSeg系统的更具启发性；相对典型部件，非典型部件对用户的启发更多。

用户测试第三阶段 在第三阶段，测试者被要求完成一份调查问卷。该调查问卷包括10个陈述句，测试者需要在标准5分李克特量表上为每个陈述打分。图20给出调查问题统计结果。总体来看，测试者对Ours系统给予较高评价。他们觉得Ours系统建议的部件更具启发性，尤其是那些非典型部件。他们认为我们的系统对造型过程的概念设计阶段具有很大助益。

与现有的要求预分割及标记的部件建议技术 [XXM⁺13, FWX⁺13] 相对，我们的技术不仅可以提供非典型部件，还能提供来自不同类模型的部件。如果允许从不同类模型内搜索部件，这类技术也可以提供来自不同类模型的部件。然而，他们仍然不太适合提供非典型部件。

建议由一组预分割部件构成的部件看起来容易，实际非常困难。这本质是个组合搜索问题。该问题的复杂度随预分割

Table 5: 各分值内非典型部件与典型部件所占比例统计结果。“Ours::Atypical”与“Ours::Regular”分别表示Ours系统的造型结果内非典型部件与典型部件在各分值内所占比例。“PreSeg::Atypical”与“PreSeg::Regular”分别表示PreSeg系统的造型结果内非典型部件与典型部件在各分值内所占比例。

Type \ Grade	1	2	3	4	5
Ours::Atypical	0.0	0.08	0.03	0.6	0.91
Ours::Regular	1.0	0.92	0.97	0.4	0.09
PreSeg::Atypical	0.0	0.0	0.0	0.0	0.0
PreSeg::Regular	1.0	1.0	1.0	1.0	1.0

部件的增长而急剧增长。从某种程度上讲，我们的RC- k NNG技术就是针对组合搜索问题的。如果现有的部件建议技术 [CK10, XXM⁺13, FWX⁺13]采用我们的RC- k NNG技术，他们也可以达到类似我们技术的效果。

8 本章小结

在本章中，我们提出针对浏览式草图造型的按需部件建议技术。在用户输入草图表示其造型意图后，我们的系统基于草图得到三维代理部件，并以之为查询模型到三维模型数据库内搜索候选部件。这些候选部件是实时地从数据库模型上匹配并分割下来的。与以前的方法不同，我们的方法不需要预分割的数据库，而且是实时地提取出与用户草图相匹配的部件，因此极大地扩展了建模空间。我们的系统为艺术家提供了完成开放式造型任务的新方法。同时，我们的方法也推动了草图式造型技术的发展。

References

- [AI08] Alexandr Andoni and Piotr Indyk. Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. *Communications of the ACM*, 51(1), 2008.
- [ATOM07] Andrew Nealen, Takeo Igarashi, Olga Sorkine, and Marc Alexa. FiberMesh: designing freeform surfaces with 3D curves. *ACM Transactions on Graphics*, 26(3):Article No. 41, 2007.
- [AZC⁺12] O.K.-C. Au, Youyi Zheng, Menglin Chen, Pengfei Xu, and Chiew-Lan Tai. Mesh segmentation with concavity-aware fields. *IEEE Transactions on Visualization and Computer Graphics*, 18(7):1125–1134, 2012.
- [BK04] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, 2004.
- [CK10] Siddhartha Chaudhuri and Vladlen Koltun. Data-driven suggestions for creativity support in 3d modeling. *ACM Transacitons on Graphics*, 29(6):Article No. 183, 2010.
- [CKGK11] Siddhartha Chaudhuri, Evangelos Kalogerakis, Leonidas Guibas, and Vladlen Koltun. Probabilistic reasoning for assembly-based 3d modeling. *ACM Transacitons on Graphics*, 30(4):Article No. 35, 2011.
- [CTSO03] Ding-Yun Chen, Xiao-Pei Tian, Yu-Te Shen, and Ming Ouhyoung. On visual similarity based 3D model retrieval. *Computer Graphics Forum*, 22(3):223–232, 2003.
- [DFRS03] Doug DeCarlo, Adam Finkelstein, Szymon Rusinkiewicz, and Anthony Santella. Suggestive contours for conveying shape. *ACM Transactions on Graphics*, 22(3):848–855, 2003.
- [ERB⁺12] Mathias Eitz, Ronald Richter, Tammy Boubekeur, Kristian Hildebrand, and Marc Alexa. Sketch-based shape retrieval. *ACM Transactions on Graphics*, 31(4), 2012.
- [FKS⁺04] Thomas Funkhouser, Michael Kazhdan, Philip Shilane, Patrick Min, William Kiefer, Ayellet Tal, Szymon Rusinkiewicz, and David Dobkin. Modeling by example. *ACM Transactions on Graphics*, 23(3):652–663, 2004.
- [FWX⁺13] Lubin Fan, Ruimin Wang, Linlin Xu, Jiansong Deng, and Ligang Liu. Modeling by drawing with shadow guidance. *Computer Graphics Forum*, 32(7):157–166, 2013.
- [GF08] Aleksey Golovinskiy and Thomas Funkhouser. Randomized cuts for 3D mesh analysis. *ACM Transactions on Graphics*, 27(5), 2008.
- [HKG11] Qixing Huang, Vladlen Koltun, and Leonidas Guibas. Joint shape segmentation with linear programming. *ACM Transactions on Graphics*, 30(6):125:1–125:12, December 2011.
- [JLCW06] Zhongping Ji, Ligang Liu, Zhonggui Chen, and Guojin Wang. Easy mesh cutting. *Computer Graphics Forum*, 25(3), 2006.
- [KT03] Sagi Katz and Ayellet Tal. Hierarchical mesh decomposition using fuzzy clustering and cuts. *ACM Transacitons on Graphics*, 22(3), 2003.
- [LF08] Jeehyung Lee and Thomas Funkhouser. Sketch-based search and composition of 3d models. In *Proceedings of the Fifth Eurographics Conference on Sketch-Based Interfaces and Modeling*, SBM’08, pages 97–104, Aire-la-Ville, Switzerland, Switzerland, 2008. Eurographics Association.

- [LIAL07] Joseph J. LaViola, Takeo Igarashi, Christine Alvarado, and Hod Lipson. Sketch-based interfaces: techniques and applications. In *SIGGRAPH Course Notes*, 2007.
- [ML14] M. Muja and D.G. Lowe. Scalable nearest neighbor algorithms for high dimensional data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(11), 2014.
- [RDB10] Hayko Riemenschneider, Michael Donoser, and Horst Bischof. *Computer Vision – ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5–11, 2010, Proceedings, Part V*, chapter Using Partial Edge Contour Matches for Efficient Object Category Localization, pages 29–42. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.
- [Sha08] Ariel Shamir. A survey on mesh segmentation techniques. *Computer Graphics Forum*, 27(6):1539–1556, 2008.
- [TSH99] Takeo Igarashi, Satoshi Matsuoka, and Hidehiko Tanaka. Teddy: a sketching interface for 3D freeform design. In *Proceedings of ACM SIGGRAPH*, pages 409–416, 1999.
- [WWZ⁺12] Jing Wang, Jingdong Wang, Gang Zeng, Zhuowen Tu, Rui Gan, and Shipeng Li. Scalable k-nn graph construction for visual descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1106–1113. IEEE, 2012.
- [XXM⁺13] Xiaohua Xie, Kai Xu, Niloy J. Mitra, Daniel Cohen-Or, Wenyong Gong, Qi Su, and Baoquan Chen. Sketch-to-design: Context-based part assembly. *Computer Graphics Forum*, 32(8):233–245, 2013.



Figure 11: 系统造型结果的例子。在每张子图内，我们给出用户绘制的草图（紫色），生成的代理模型（浅蓝色），对应的数据库模型及提取出的部件（数据库模型上的蓝色部件）。注意结果模型都是由非常规部件或来自不同类的部件构成。

a



b



c



Figure 12: 用我们系统设计的模型：半人马怪物玩具（a），三维打印台灯（b），三维打印乌龟怪兽（c）。

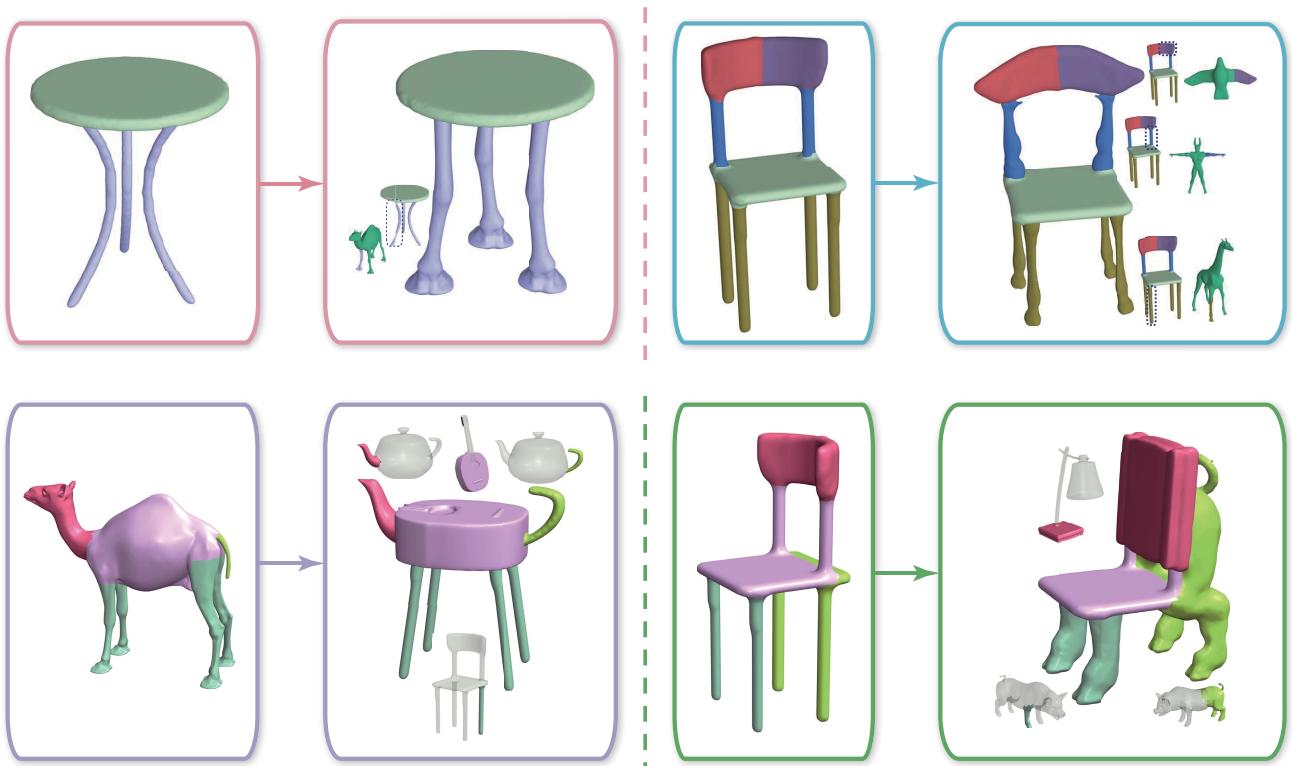


Figure 13: 模型变异的例子。输入一个分割后的模型（左侧），以其部件为输入去数据库内搜索相似部件，得到与输入模型结构类似但外形明显不同的模型。

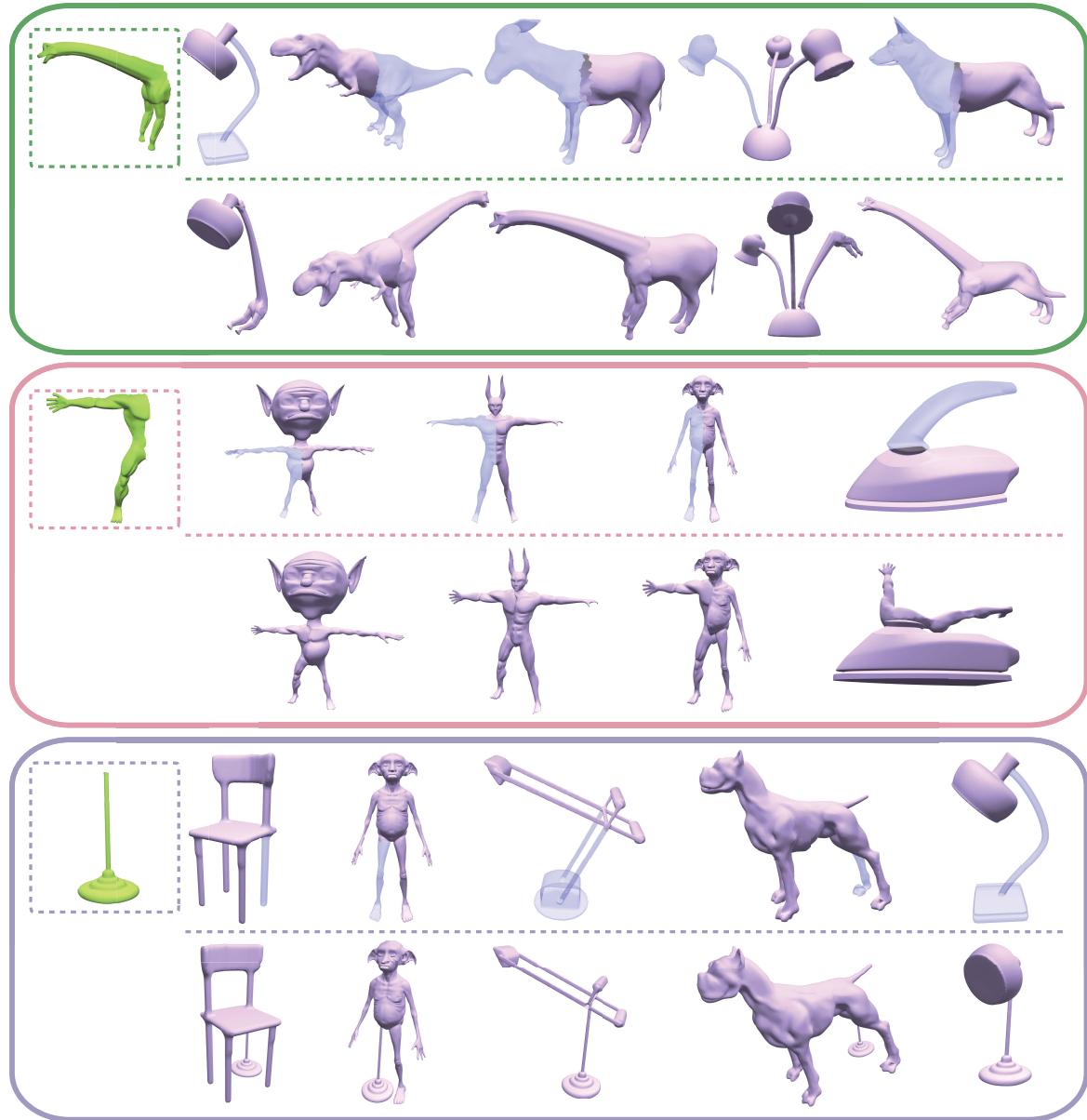


Figure 14: 部件建议的例子。输入一个查询模型（绿色），我们的系统给出一系列建议部件（紫色）。组合后的模型在建议部件的下方。

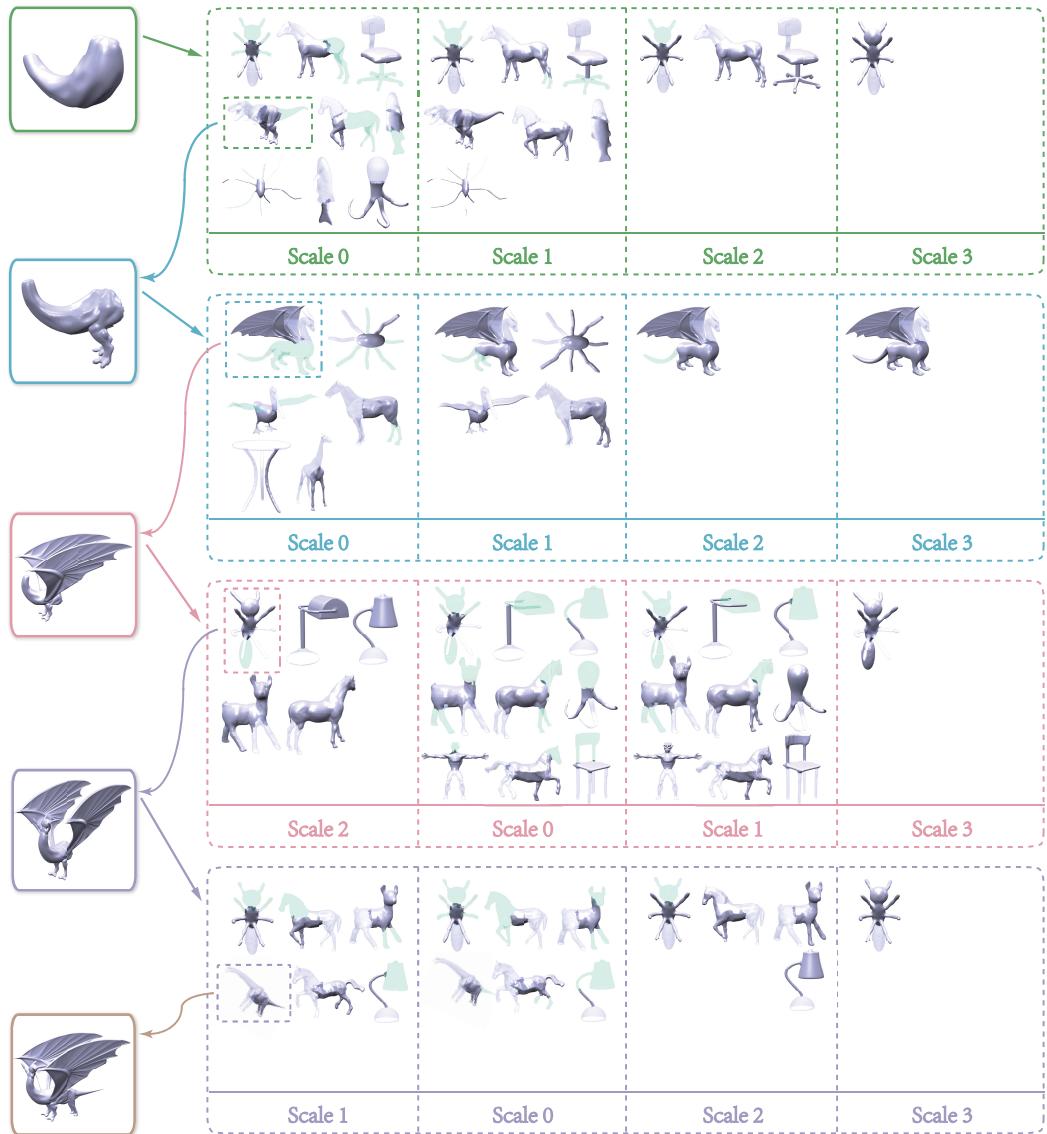


Figure 15: 多尺度部件建议的例子。初始模型及每个步骤得到的结果模型，做为查询部件，列在左侧。在右侧，我们给出每步查询得到的不同尺度下的建议部件（红色）。数据库模型上与查询部件相对应的部分标为浅蓝色。其它部件标为浅绿色。

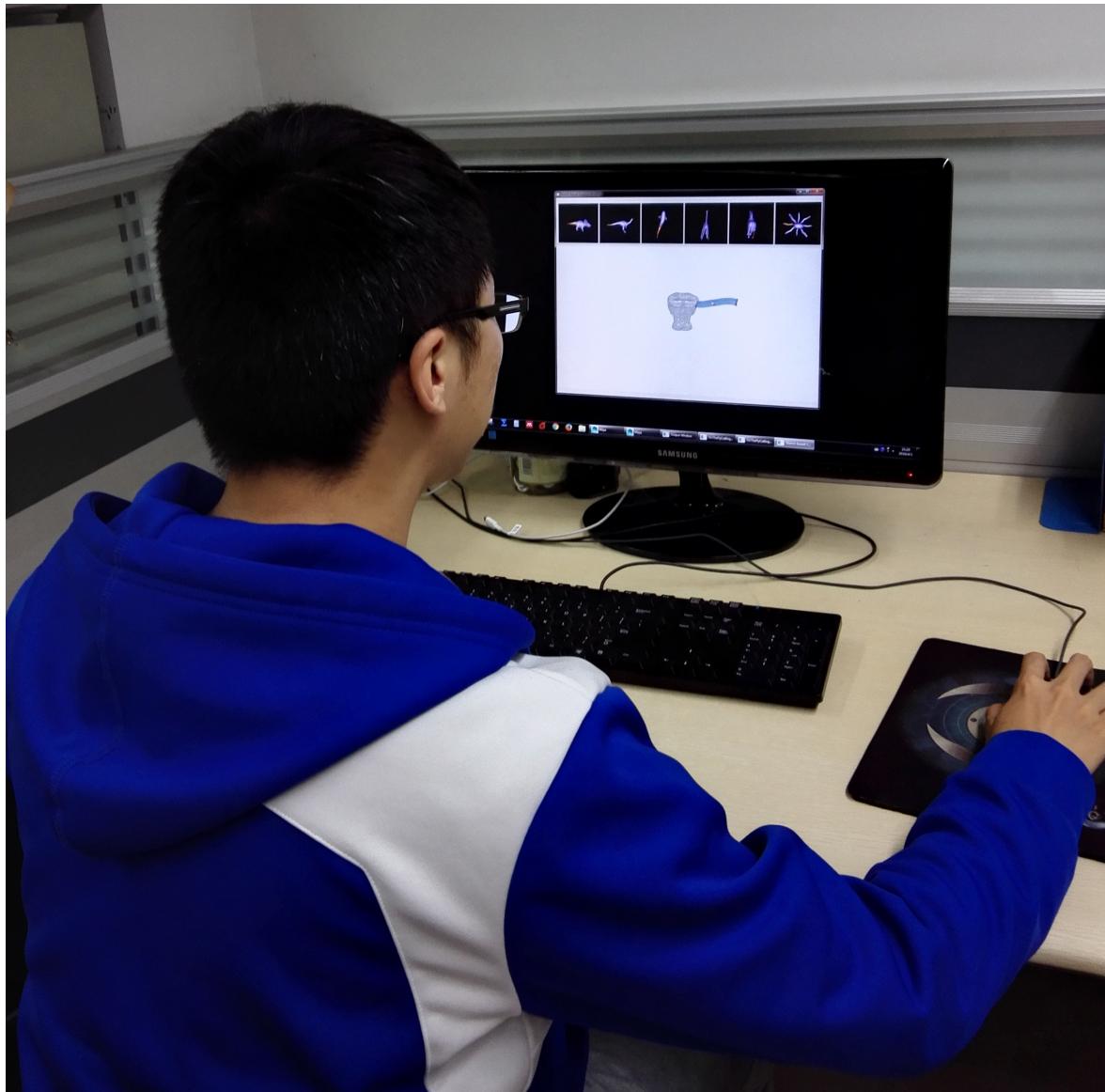


Figure 16: 一位测试者正在操作造型系统完成造型任务。

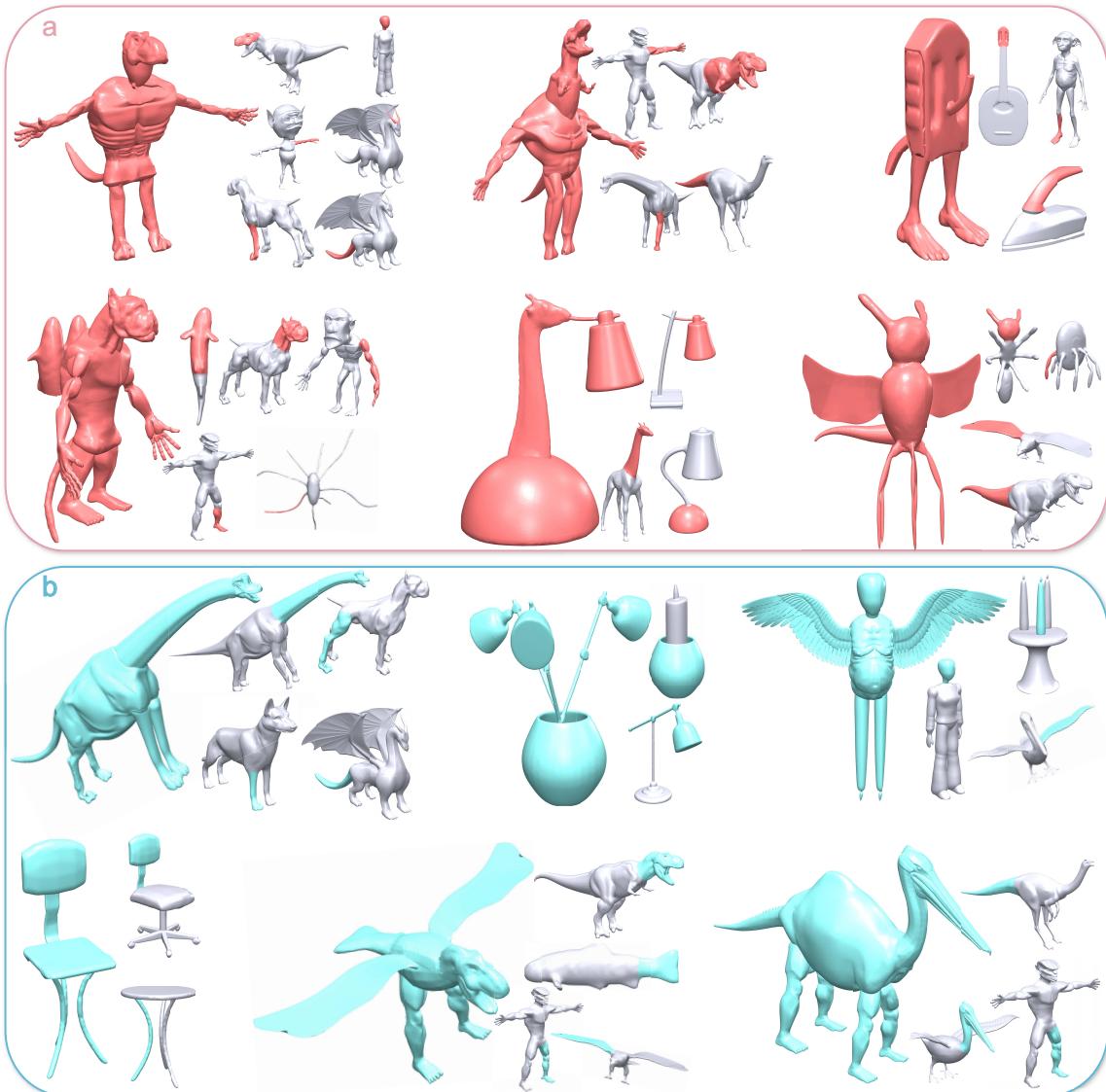


Figure 17: Ours系统 (a) 与PreSeg系统 (b) 造型结果。每个造型结果，其构成部件在右侧。

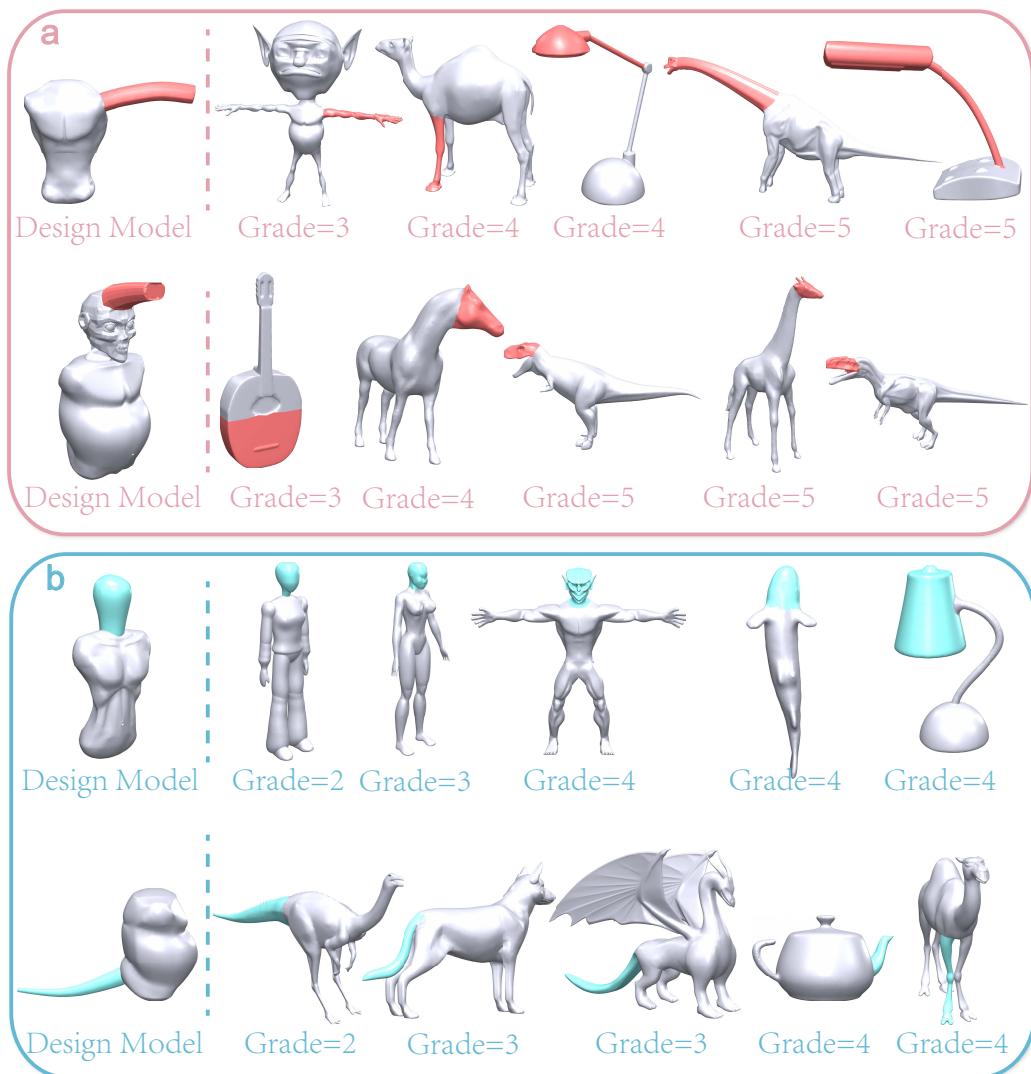


Figure 18: 具有代表性的建议部件及其得分。 (a) 图给出Ours系统建议的部件（红色）。 (b) 图给出PreSeg系统建议的部件（蓝色）。

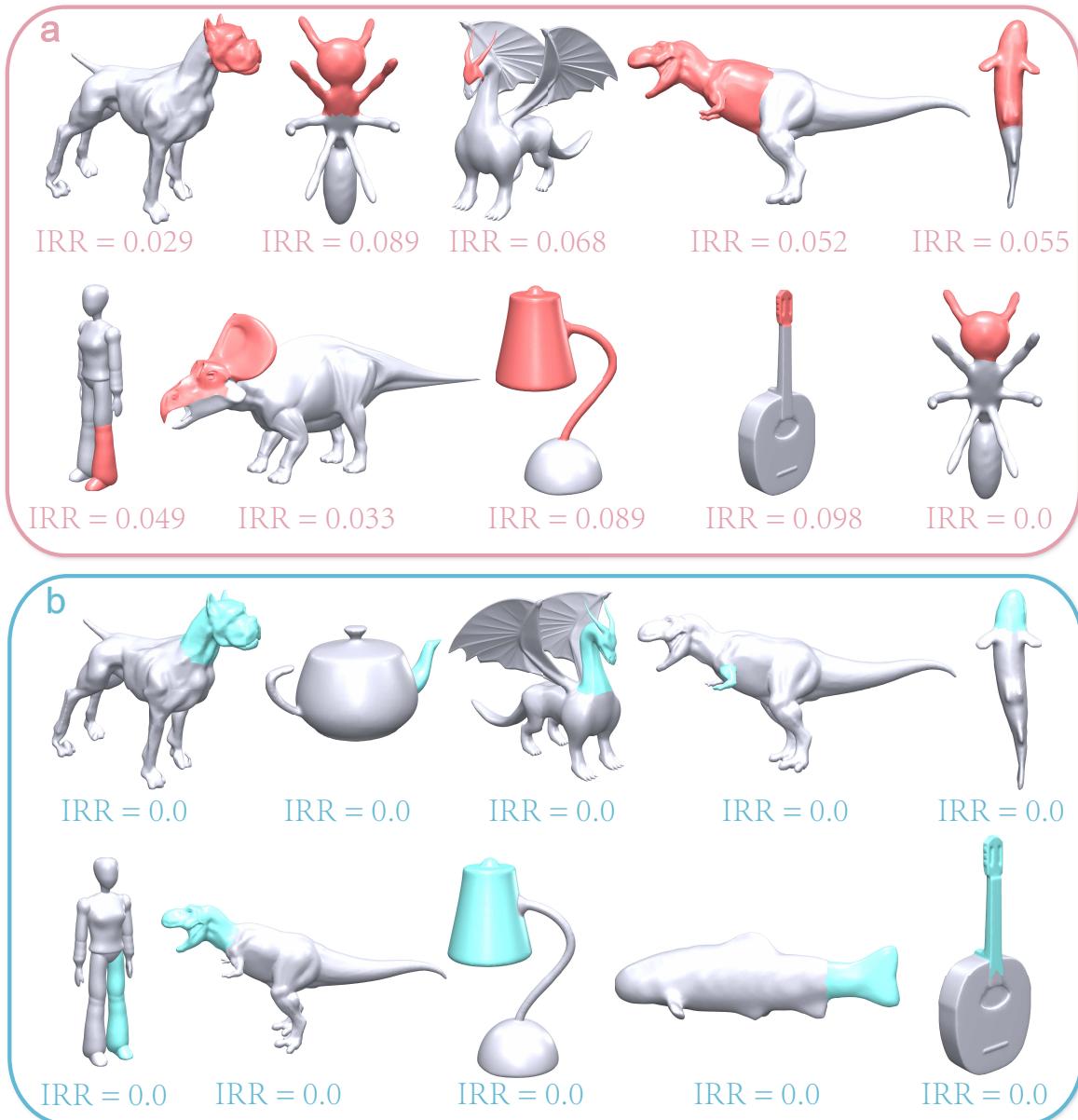


Figure 19: 具有代表性的建议部件及其非规则度量值。 (a) 图给出Ours系统建议的部件（红色）。 (b) 图给出PreSeg系统建议的部件（蓝色）。

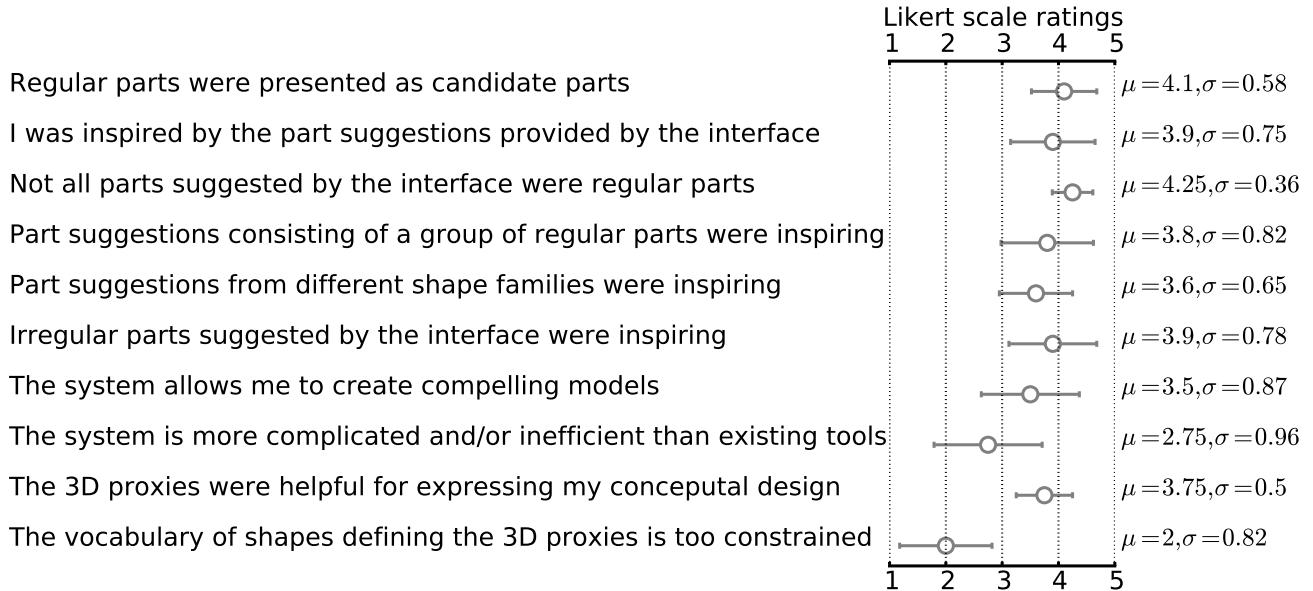


Figure 20: 调查问卷统计结果。“1”表示“强烈同意”。“2”表示“同意”。“3”表示“中立”。“4”表示“不同意”。“5”表示“强烈不同意”。