Shape2Motion: Joint Analysis of Motion Parts and Attributes from 3D Shapes Supplemental Material

Xiaogang Wang[1] Bin Zhou[1] Yahao Shi[1] Xiaowu Chen[1] Qinping Zhao[1] Kai Xu[2*]
[1]Beihang University [2]National University of Defense Technology

1. Introduction

This supplemental material contains six parts:

- Section 2 provides an overview and detailed statistics of the Shape2Motion benchmark dataset.
- Section 3 gives more details on the annotation tool used to label our benchmark data.
- Section 4 provides the details on training data generation and augmentation.
- Section 5 shows the baseline network.
- Section 6 gives a few representative failure cases along with discussions.
- Section 7 shows more results of mobility analysis on the Shape2Motion benchmark.

2. Shape2Motion benchmark dataset

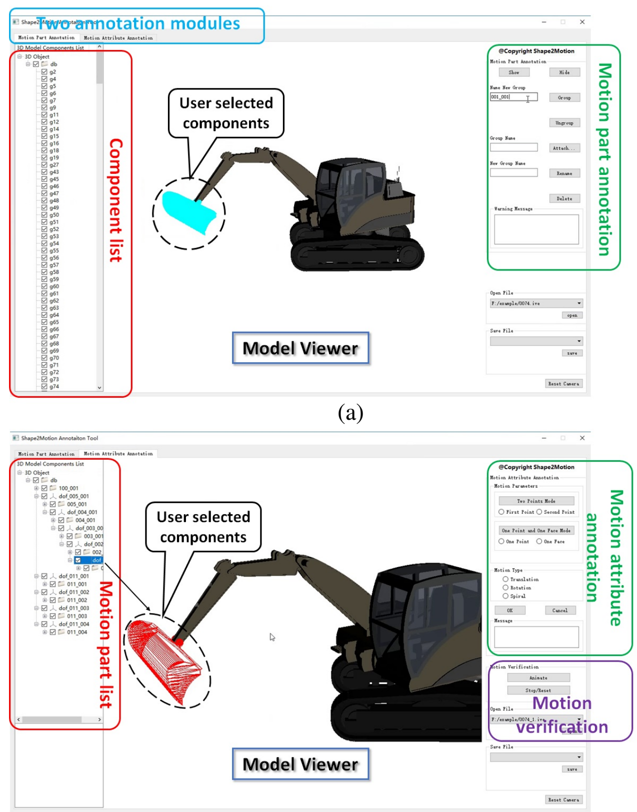
Table 1 reports the detailed statistics of the benchmark dataset. An overview of the benchmark dataset (45 categories) are given in Figure 7, 8, 9, 10, and 11.

3. Annotation tool

The interactive annotation tool consists of two pars, one for motion part annotation and one for motion attribute annotation. For a given 3D model, our annotation process is split into two stages: 1) Motion parts annotation (Figure 1 (a)); 2) Motion attribute annotation (Figure 1(b)).

For motion part annotation (Figure 1 (a)), we first load a 3D model file and extract all components from the model. All components are shown in the component list (red). The user can select all components belonging to one motion part from the main viewer or from the components list.

In the second stage, the user picks a motion part from the motion part list (red) and annotates its motion attribute (Figure 1 (b)). As shown in the motion attribute annotation region (green), motion attribute annotation is composed of



(a)

(b)

Figure 1: GUI for our mobility annotation tool. The tool consists of two parts, one for motion part annotation (a) and one for motion attribute annotation (b).

two parts: motion parameters and motion type. For motion parameters annotation, we offer two modes for the convenience of interactive annotation: 1) two-point mode and 2) point+face mode. For motion type annotation, the user can simply choose one of the motion types.

To facilitate the user to visually verify the correctness of a annotated mobility, we develop an animation-based motion verification function. This is achieved by animating the annotated motion part with the corresponding motion attributes prescribed by the user, as shown in the motion

*Corresponding author: kevin.kai.xu@gmail.com

Category Name	Bicycle	Folding Chair	Motorbike	Rocking Chair	Swivel Chair	Vehicle	Bottle	Bucket	Cabinet
# Models	63	21	107	19	21	101	56	14	30
# Motion Parts	256	42	321	19	42	1101	66	16	108
# Avg. Motion Parts	4.06	2	3	1	2	10.9	1.18	1.14	3.6
# R/T/R+T	256/0/0	42/0/0	321/0/0	19/0/0	21/21/0	1101/0/0	3/0/63	16/0/0	46/62/0
Category Name	Cannon	Carton	Clock	Door	Doorknob	Electric fan	Excavator	Faucet	Glasses
# Models	102	8	58	29	63	64	39	162	43
# Motion Parts	348	18	126	69	182	85	286	251	86
# Avg. Motion Parts	3.41	2.25	2.17	2.38	2.89	1.33	7.33	1.55	2
# R/T/R+T	348/0/0	16/2/0	126/0/0	68/1/0	182/0/0	85/0/0	286/0/0	251/0/0	86/0/0
Category Name	Globe	Handcart	Helicopter	Kettle	Lamp	Laptop	Lighter	Oven	Pen
# Models	29	121	104	61	68	86	37	42	52
# Motion Parts	32	453	202	66	175	86	66	43	53
# Avg. Motion Parts	1.1	3.74	1.94	1.08	2.57	1	1.78	1.02	1.02
# R/T/R+T	32/0/0	453/0/0	202/0/0	17/49/0	175/0/0	86/0/0	47/19/0	43/0/0	0/53/0
Category Name	Plane	Refrigerator	Revolver	Scissors	Screwdriver	Seesaw	Skateboard	Stapler	Swing
# Models	143	81	25	26	70	23	79	33	36
# Motion Parts	651	150	71	52	70	23	324	65	59
# Avg. Motion Parts	4.55	1.85	2.84	2	1	1	4.1	1.97	1.64
# R/T/R+T	651/0/0	131/19/0	71/0/0	52/0/0	70/0/0	23/0/0	324/0/0	65/0/0	59/0/0
Category Name	Knife	Tank	Toilet	Valve	Washer	Watch	Windmill	Window	Wine Bottle
# Models	17	107	64	36	62	6	78	14	17
# Motion Parts	93	268	103	36	62	15	78	27	17
# Avg. Motion Parts	5.47	4.95	1.61	1	1	2.5	1	1.93	1
# R/T/R+T	93/0/0	268/0/0	103/0/0	36/0/0	62/0/0	15/0/0	78/0/0	17/10/0	0/17/0

Table 1: Statistics of the Shape2Motion benchmark.

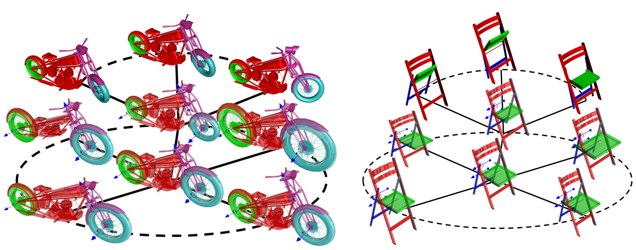


Figure 2: Training data augmentation. Given a shape (center), the shapes lying in 1-ring and 2-ring neighbor are its geometric variations and motion variations, respectively.

verification region (purple).

With our annotation tool, it takes 80 seconds in average to annotate a 3D shape for a CS graduate student user. We also submitted a supplementary video to demonstrate the complete annotation process with an example model.

4. Training data generation and enhancement

For the Motion Part Proposal Module, the ground truth similarity matrix can be directly used for training. For the Motion Axis Proposal Module, we select K anchor points for each ground truth motion axis from input point cloud, and we set K=30 in all experiments in this paper. For each anchor point, the corresponding classification and regression quantities are calculated according to ground truth motion axes, and the details are shown in Section 3.2. At the same time, the number of anchor points is much less than the number of non-critical points. In training, we adjust the training weight of anchor points to 50 for the classification task of anchor points.

For the Proposal Matching Module, we first label motion part proposals as positive if their 3D IoU scores with ground truth are larger than 0.5. Then, we evenly sample the positive proposals for each motion part of the model, total of 128 motion part proposals in each input model. If the positive samples are fewer than 128, we pad the mini-batch with the negative samples with the highest scores that from the same model.

Given a training dataset with ground-truth mobilities, we perform data augmentation via generating two kinds of shape variations. We first generate a number of *geometric variations* for each training shape based on the method described in [1]. Furthermore, based on the ground-truth mobility in the training shapes, we move the motion parts

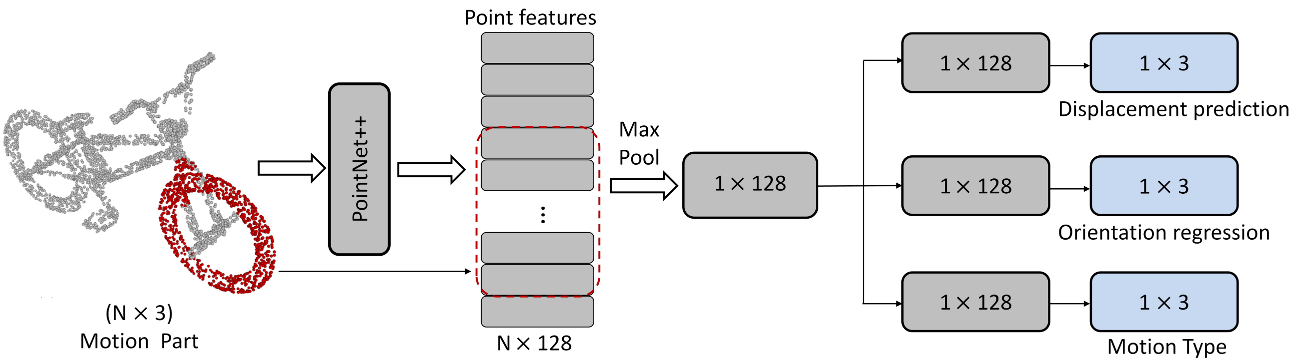


Figure 3: Baseline Network. The network takes a motion part segmentation (mask) as input. The motion part mask is applied to displacement prediction, orientation regression. The motion type branch is responsible for regressing motion parameters (anchor point, motion orientation) and motion type.

according to their corresponding motion attributes, resulting in a large number of *motion variations*. Figure 2 shows two examples of our training data augmentation.

5. Baseline Network

We design a intuitive network architecture to regress motion attribute for each motion part, see Figure 3. For a given motion part generated by SGPN [2], the network first extract features for each point from input model through a PointNet++ architecture. Then motion part mask is applied to align corresponding feature map and aggregate point features by max pooling. For each task, motion part feature is mapped to a feature vector by a fully connected layer. The network produces three output vectors per motion part.

For displacement prediction, we regress the displacement vector between the center (at the origin $(0, 0, 0)$) of the input model and the ground-truth axis. For orientation regression, the network estimates a residual vector to the ground-truth orientation. The loss is defined as:

$$L = L_{\text{type}} + L_{\text{dis}} + L_{\text{res}}, \tag{1}$$

where L_{type} is a softmax classification loss, which is trained to classify the mobility into one of the three motion types. L_{res} and L_{dis} take L2 loss between the predicted displacement/orientation and ground-truth.

6. Failure cases

Figure 4 shows two classes representative failure cases of our method. The ground-truth mobility and the prediction are given in the left and right. In the first case (Figure 4(a)), the tiny motion parts (the hands of a watch or a clock) are completely missing. In the second case (Figure 4(b)), our method fails in models with severely crossed or nested structure, such as blade/scabbard, injector, etc. The reason is that point clouds lose a lot of geometric detail.

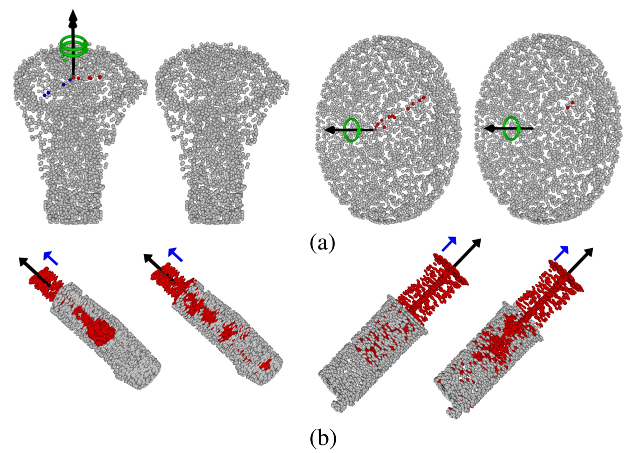


Figure 4: Failure cases. (a): The tiny motion parts (the hands of a watch or a clock) are missing. (b): Our method fails in models with severely crossed or nested structure, such as the blade/scabbard and the injector.

7. More results of mobility analysis

Figure 5 and 6 provide more results of mobility analysis on the diverse collection of shapes from our Shape2Motion benchmark.

References

[1] Q. Fu, X. Chen, X. Su, J. Li, and H. Fu. Structure-adaptive Shape Editing for Man-made Objects. *Computer Graphics Forum*, 2016. 2

[2] W. Wang, R. Yu, Q. Huang, and U. Neumann. SGPN: similarity group proposal network for 3d point cloud instance segmentation. *CoRR*, abs/1711.08588, 2017. 3

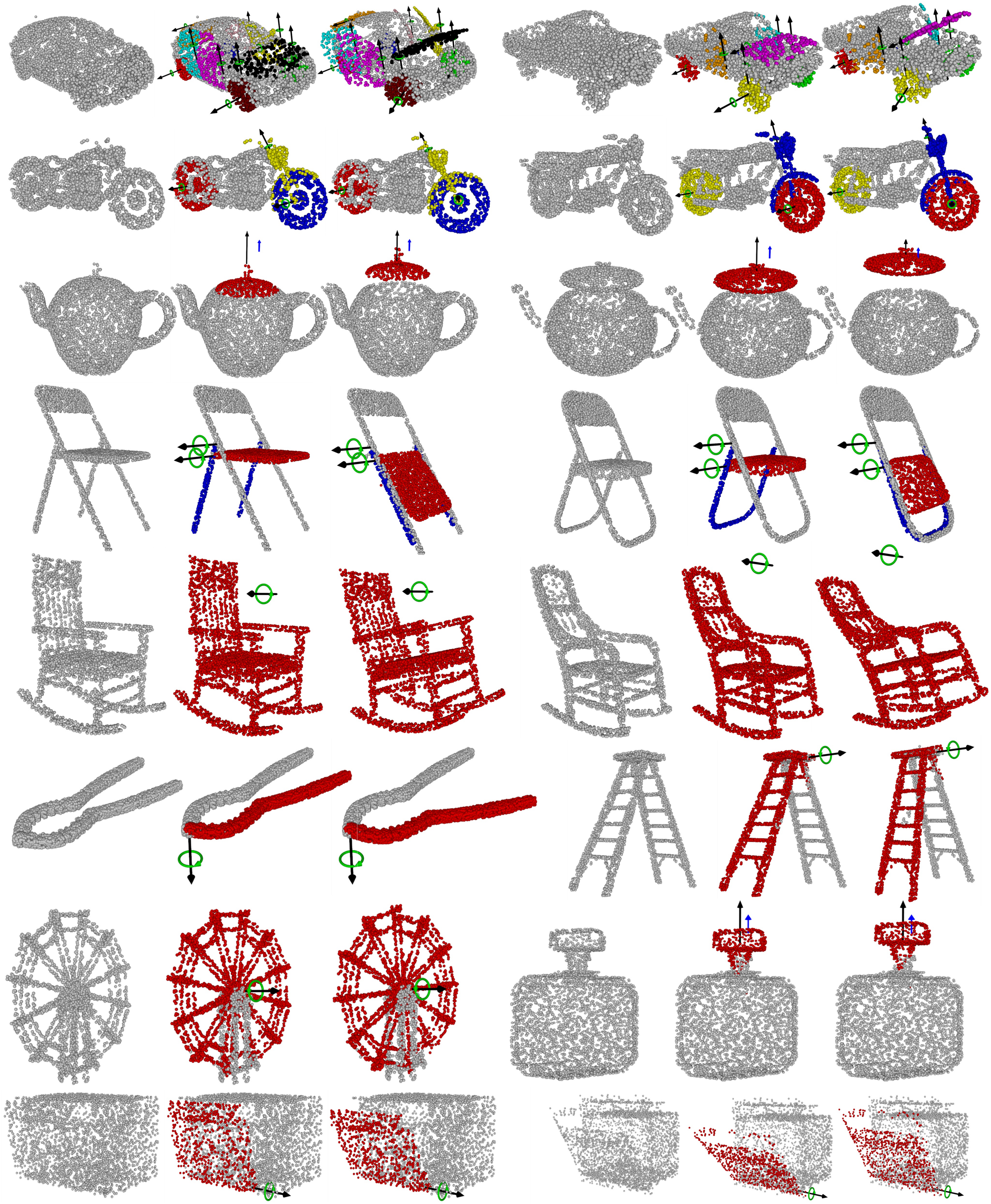


Figure 5: More results of mobility analysis.

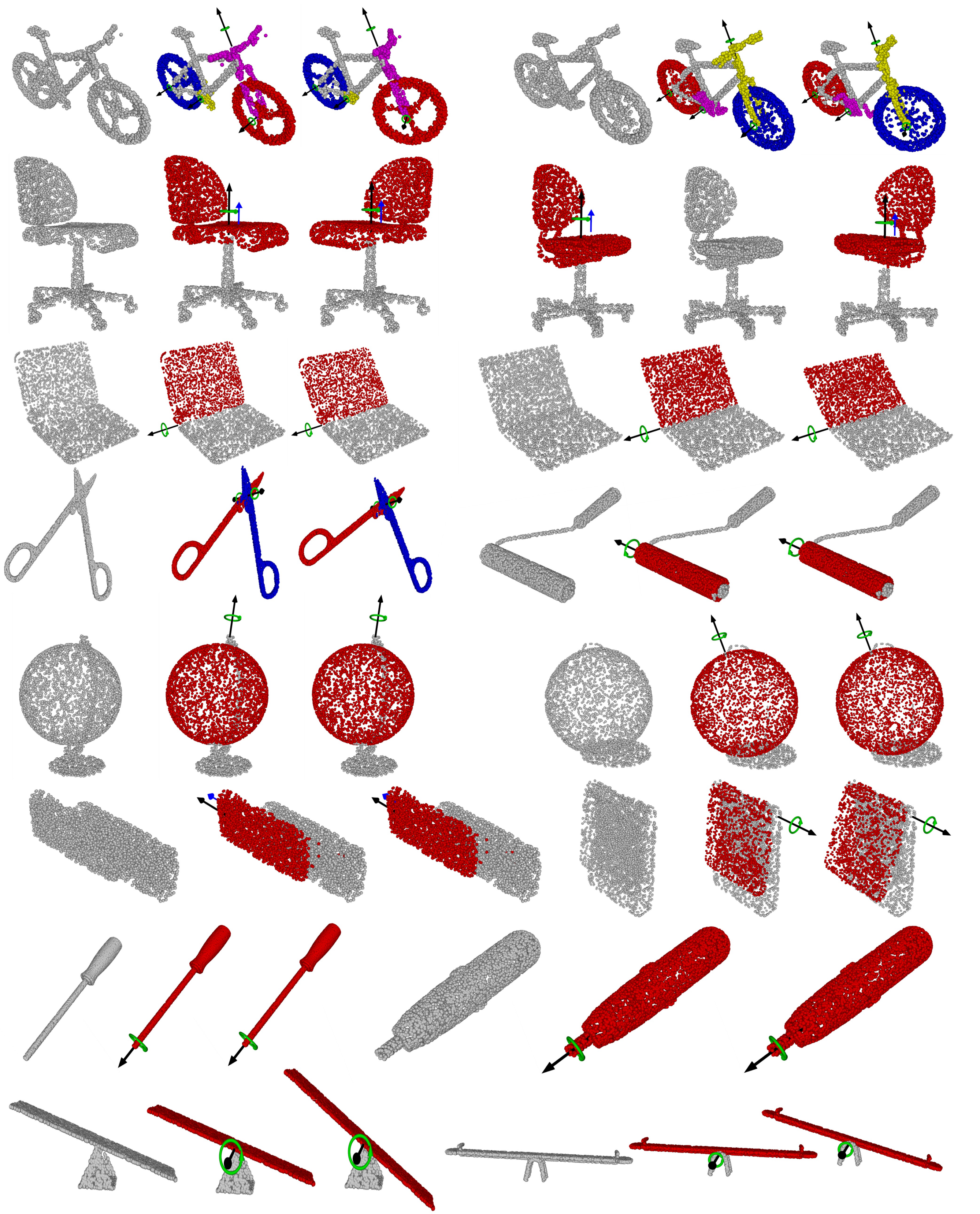


Figure 6: More results of mobility analysis.

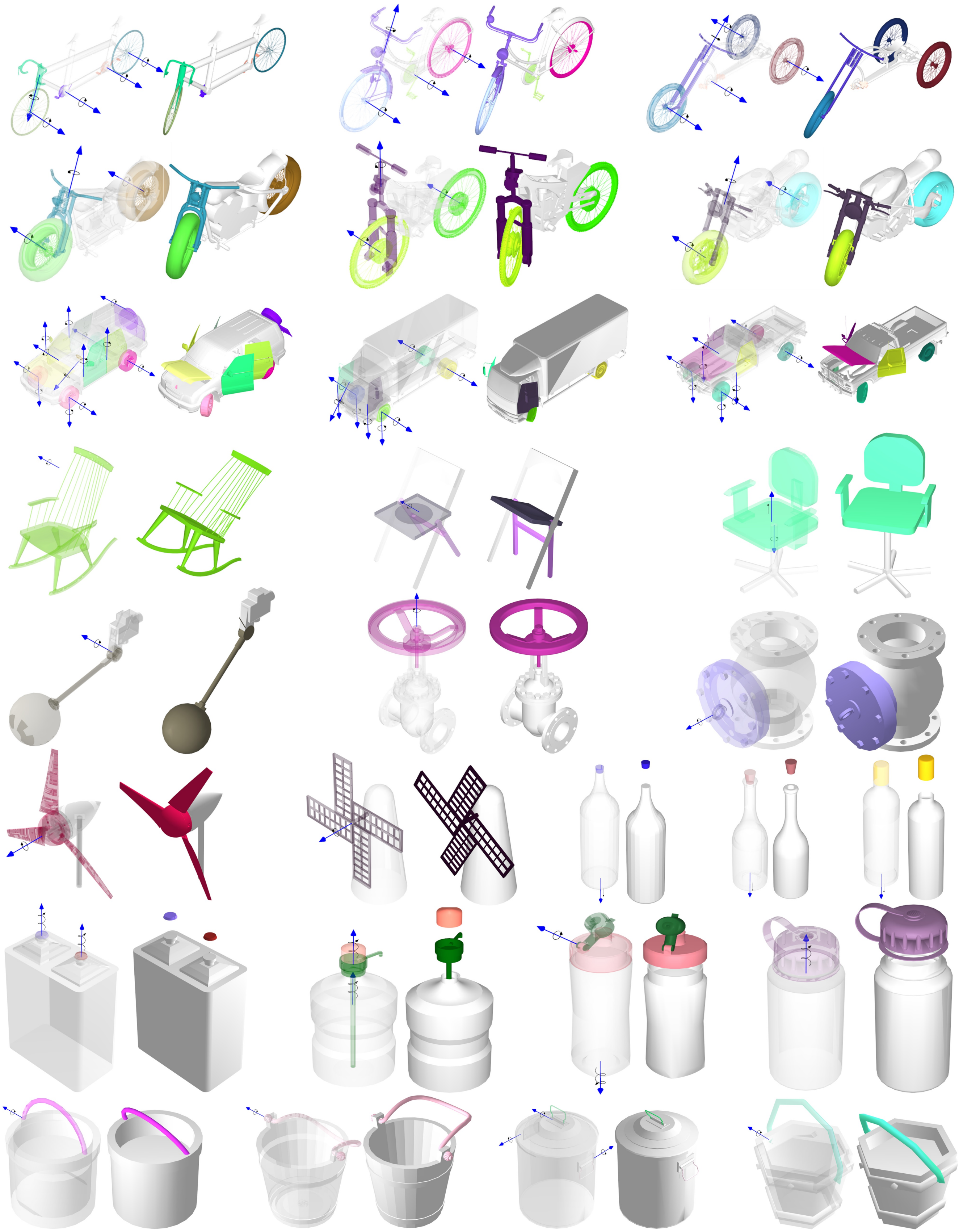


Figure 7: Overview of the Shape2Motion benchmark.

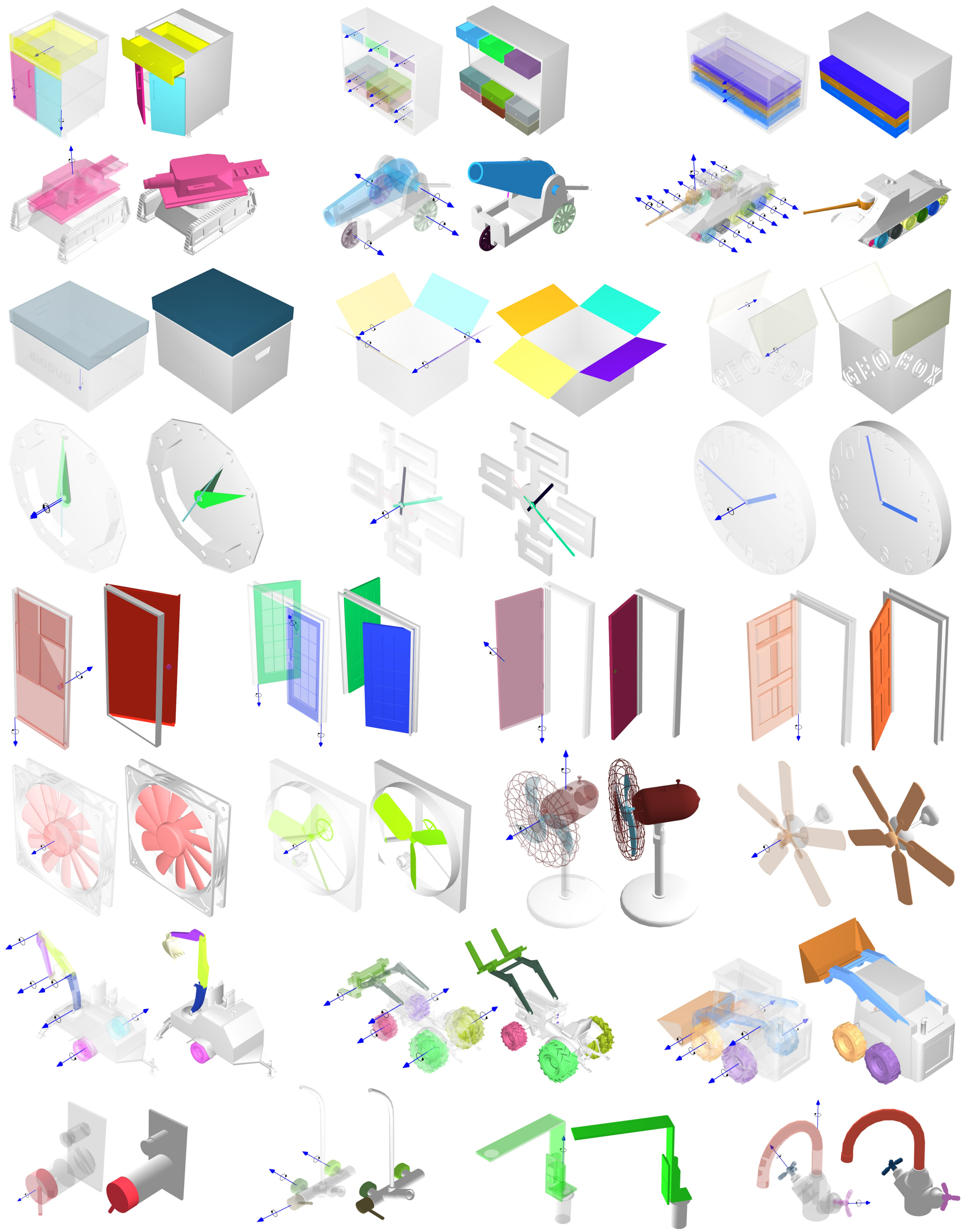


Figure 8: Overview of the Shape2Motion benchmark.

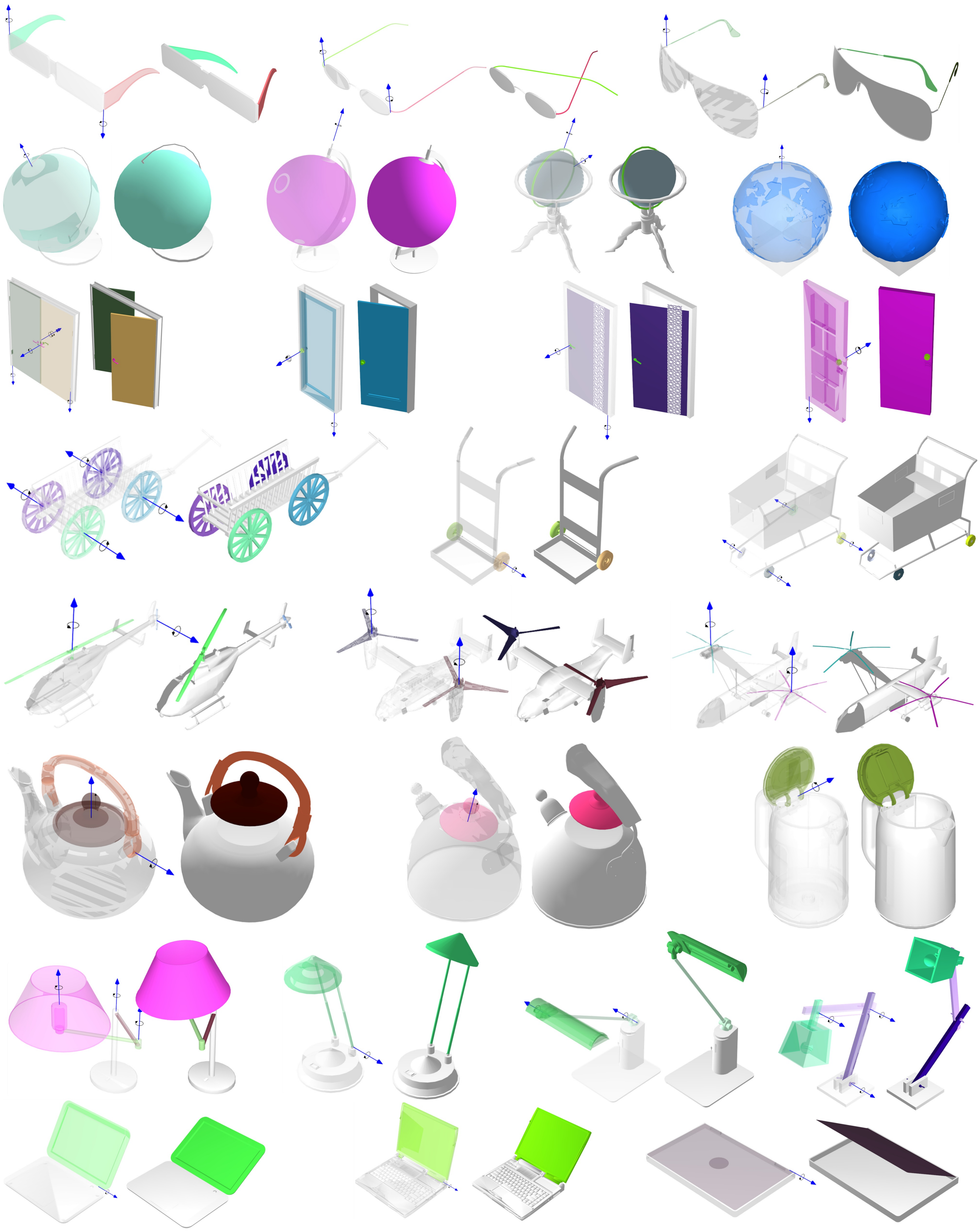


Figure 9: Overview of the Shape2Motion benchmark.

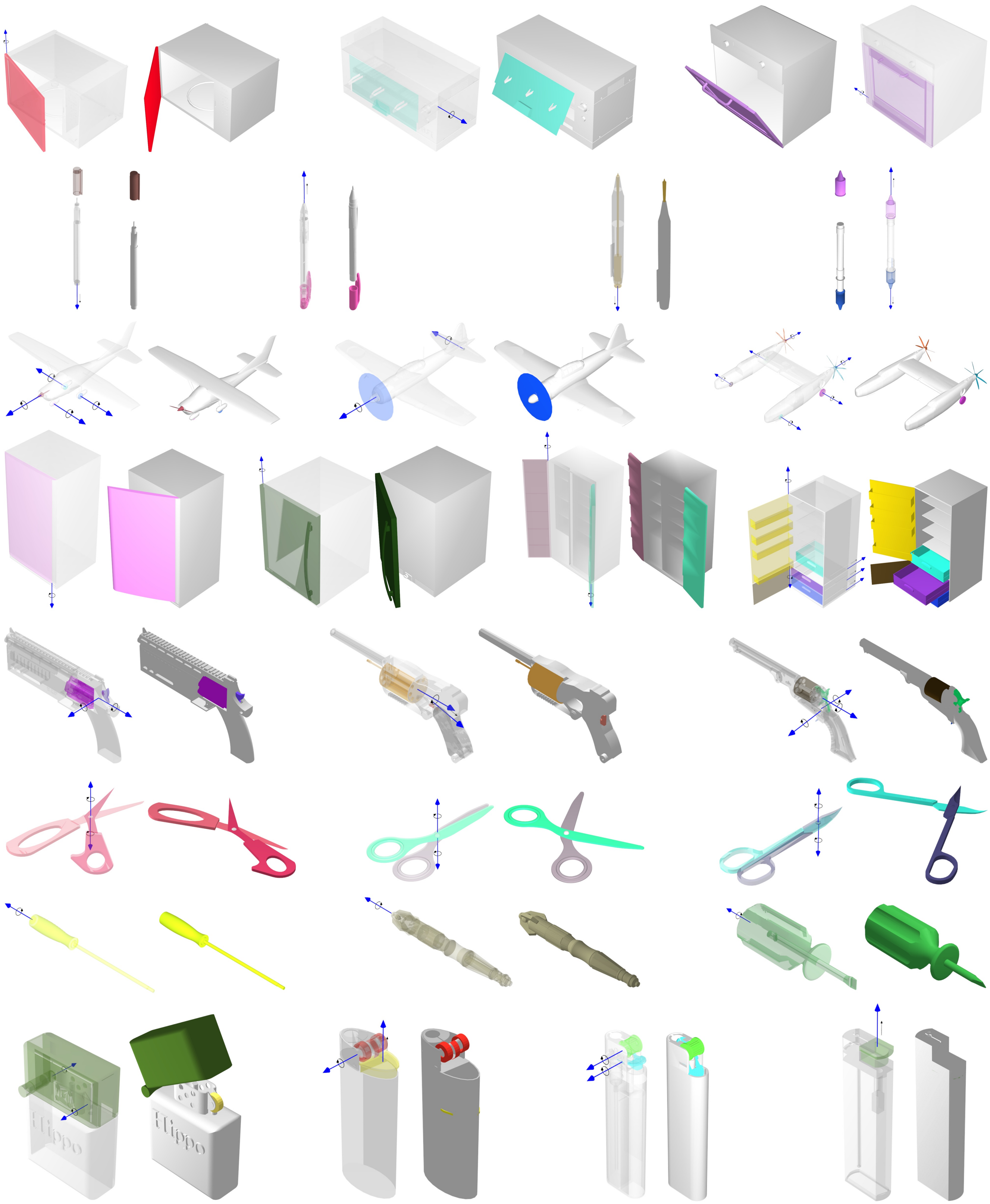


Figure 10: Overview of the Shape2Motion benchmark.

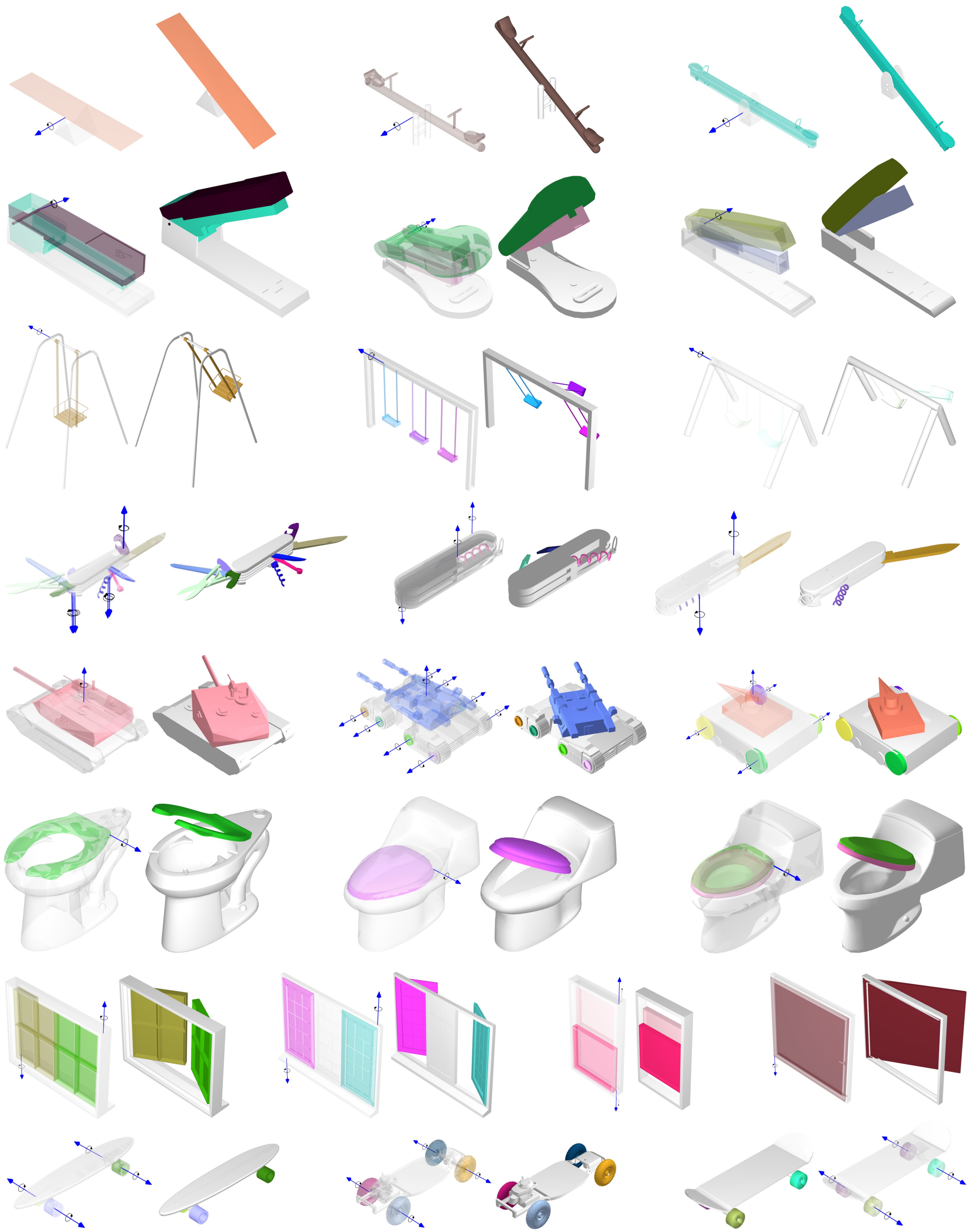


Figure 11: Overview of the Shape2Motion benchmark.