On Being the Right Scale: Sizing Large Collections of 3D Models

Manolis Savva, Angel X. Chang, Gilbert Bernstein, Christopher D. Manning, Pat Hanrahan
Computer Science Department, Stanford University

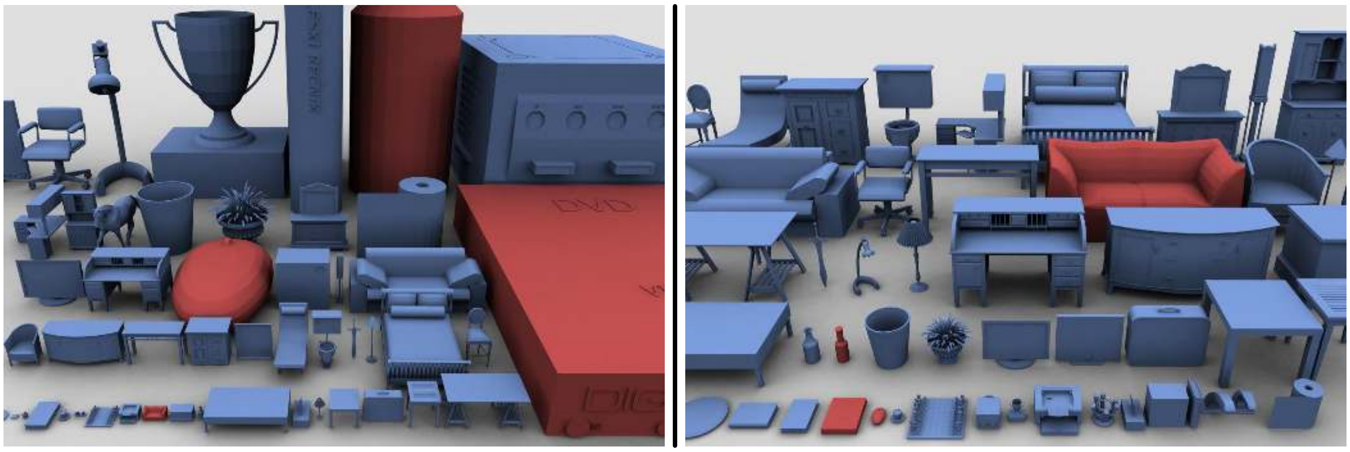


Figure 1: *We use a probabilistic model to determine scales that give a collection of 3D models physically plausible sizes. Left: fifty models randomly sampled from a 12490 model dataset. Many sizes are implausible (red highlights some particularly implausible cases). Right: same models rescaled with our approach (e.g. the DVD player and computer mouse are now plausibly sized in the front row on the right).*

Abstract

We address the problem of recovering reliable sizes for a collection of models defined using scales with unknown correspondence to physical units. Our algorithmic approach provides absolute size estimates for 3D models by combining category-based size priors and size observations from 3D scenes. Our approach handles unobserved 3D models without any user intervention. It also scales to large public 3D model databases and is appropriate for handling the *open-world problem* of rapidly expanding collections of 3D models. We use two datasets from online 3D model repositories to evaluate against both human judgments of size and ground truth physical sizes of 3D models, and find that an algorithmic approach can predict sizes more accurately than people.

Keywords: 3D models, automatic scaling, probabilistic modeling, data-driven methods

1 Introduction

Today, there are more 3D models freely and publicly available than ever before. And tomorrow, there'll be more. Databases like TurboSquid, Archive3D, and the Trimble 3D Warehouse (formerly Google 3D Warehouse) are growing every day.

These 3D model repositories are useful for a wide diversity of applications. Using these models, novices can create game levels, virtual movie sets, tell stories, and explore home remodeling. Beyond computer graphics, these databases are also used in computer vision and robotics for research problems such as object recognition and 3D scene reconstruction [Zia et al. 2011; Satkin et al. 2012].

Unfortunately, the same process driving the growth of these databases—aggregation of models from a wide variety of sources—also results in collections with poor and unreliable metadata. While poor metadata is prevalent in many domains, the problem of inconsistent sizes is unique to 3D model databases. More specifically, the correspondence between the virtual unit scale of a 3D model and physical units is typically unknown or unreliable. In contrast, physical objects possess absolute, fixed sizes.

Some 3D models come with virtual to physical unit conversion metadata embedded (notably the COLLADA spec [2008] provides a field for this purpose). Sometimes this information can be trusted because it comes from sources such as a furniture manufacturer, who have a vested interest in its correctness. In general, however, these metadata fields are unpopulated or of unknown quality.

Unreliable model sizes produce a variety of second order problems. In interactive modeling, users are burdened with the responsibility of rescaling models inserted into a scene. Due to unreliable size information, object recognition systems using 3D models typically normalize model sizes. Disregarding absolute size creates confusion between many categories of objects where absolute size is discriminative, for example thimbles and waste baskets [Wohlkinger et al. 2012]. By providing reliable physical sizes for 3D models in large repositories, we can aid future research and help automated or interactive modeling systems to use absolute size information.

To address this problem, we size models using a probabilistic model that combines category-based physical size priors and model observations in 3D scenes. Knowing the category of an object gives us a prior distribution for its size. For instance, even though databases tend to contain many diverse chair models, the size of any given chair is heavily constrained by the fact that it is a chair. By using the geometry and text of 3D models on online repositories, we can predict category membership and thus estimate sizes without requiring any user feedback. This enables us to automatically scale our method to very large databases.

Indoor Scene Understanding Where Graphics Meets Vision'14, Dec. 03-06 2014, Shenzhen, China. Copyright © ACM 978-1-4503-3242-2/14/12..$15.00
http://dx.doi.org/10.1145/2670291.2670295

To use category-based size knowledge we need to have data for the categories we want to handle. However, manual collection of this data is not a scalable solution. We show how a small set of category size priors can be automatically collected from external information sources, connected to 3D models and expanded to cover additional categories. In order to expand these priors, we use the key insight that 3D scenes serve as a link between co-occurring 3D models. Scenes containing models from our database provide observations of relative sizes between any two models. Given these observations and absolute size values or priors for a subset of models in a scene, we use the network of relative size observations to propagate size knowledge to other models.

Contributions To our knowledge, we are the first to pose absolute scaling of 3D models as a research problem. We present a probabilistic graphical model formalization of the problem and demonstrate a series of methods that determine physically plausible scales for models in 3D model databases. We show how these methods integrate different sources of size information: known size reference models, observations from 3D scenes and category-based absolute size priors. We analyze the performance of these different sources of size data independently and in combination for predicting 3D model scales. We evaluate against human size judgments and against manually annotated ground truth size models. Finally, we provide our learned size priors and best size estimates from two large 3D model datasets for the benefit of the research community.

2 Background

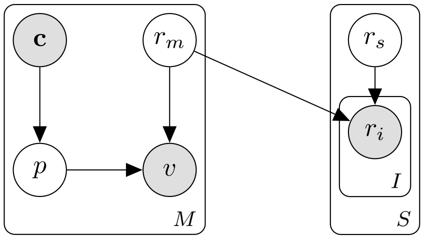
Computer vision research has recently shown that 3D model databases can be used for object categorization, and pose and depth estimation [Zia et al. 2011; Wohlkinger et al. 2012]. The authors observe that size data is critical for such tasks but usually unavailable. In the absence of reliably sized 3D model data, vision researchers have resorted to estimating object sizes from 2D images and camera parameters in order to show the benefit of size information for recognition tasks [Fritz et al. 2010].

Recent work in natural language processing has automatically extracted numerical attributes from web text [Davidov and Rappoport 2010]. That system provides estimates of height and weight values for particular named entities (such as celebrities) or object classes (such as apples). This recent work motivates our focus on providing a general, scalable approach to consistently size 3D models in large public 3D databases.

Previous research related to sizing 3D models has focused just on the geometric problem of *non-homogeneous resizing* [Kraevoy et al. 2008; Wang and Zhang 2009]. Other researchers have noted that the scales of 3D models are frequently inconsistent between models and result in physically implausible sizes [Wohlkinger et al. 2012]. However, this problem only really comes to the forefront when models are used in an external context such as a 3D scene or connected to real world data. We believe absolute sizing of 3D models has not been posed as a research problem until now because large, publicly available 3D model datasets that enable 3D model re-use and re-combination are a fairly recent phenomenon.

Thus far, work on systems that leverage absolute size information from 3D models is typically restricted to smaller, manually validated datasets [Shao et al. 2012]. However, manual validation is not a scalable solution, nor can it be applied to growing model databases. We thus target our approach to large, unstructured and unclean collections of 3D models.

Previous work has dealt with height estimation of annotated objects in collections of photographs [Hoiem et al. 2006; Lalonde et al. 2007; Russell and Torralba 2009]. This work also leverages the insight that relative size observations in scenes can be used to propagate size priors between objects. Our approach is inspired by this insight as well, but in contrast to prior work, we present an overarching probabilistic model framework and evaluate our algorithm against ground truth and human judgments. We investigate the usability of results from this prior work as size priors for 3D models and find that there is limited overlap in categories and a consistent upward bias with respect to ground truth data (see Section 4.3). Furthermore, these methods rely on manually annotated and categorized 2D object instances and do not address the issue of categorization, whereas we present an unsupervised 3D model categorization algorithm. However, we believe this prior work illustrates the importance and utility of methods for collecting and propagating physical size priors. Our aim is to address the problem of sizing collections of 3D models, integrating different information sources and systematically evaluating our results against ground truth data and human judgments.



Notation	Interpretation
$\mathbf{c} \in 1 \ldots K$	Category label
$v \in \mathbb{R}^+$	Virtual units
$p \in \mathbb{R}^+$	Physical units
$r_m \in \mathbb{R}^+$	Model scale ($p = v \cdot r_m$)
$r_s \in \mathbb{R}^+$	Scene scale
$r_i \in \mathbb{R}^+$	Model instance scale

Figure 2: *Plate notation (top) and random variables (bottom) in our probabilistic graphical model for 3D model sizes. Latent variables are in white circles, observed variables are in gray. The domain of each plate is indicated by M (models), S (scenes), and I (model instances observed in scenes).*

3 Approach

Our goal is to determine scales for 3D models so that the absolute sizes of the models are plausible to human observers. We note that this does not require determination of exact sizes due to size variation of many physical objects and variation of human size judgments. Even in cases where categories of objects have standard sizes, for example soda cans, human judgment of their size can vary widely. We choose a probabilistic representation for object scales to account for such noise and variability.

Our approach is based on two key insights: (a) in the real world and in 3D scenes, co-occurrences of objects provide a dense set of relative size observations and (b) object categories generalize size knowledge and provide priors on the expected physical sizes of objects. The latter relies on a categorization of objects and size prior data for each category. The former relies on observing objects co-occurring in scenes and having grounding data to convert relative sizes to absolute sizes. This grounding data can either be a subset of the observed models with known absolute sizes or category-based priors from part (b).

We present algorithms that use these two insights independently and an approach combining both. First, we formalize our problem with a probabilistic model (Figure 2). The left side of the plate diagram corresponds to category-based size priors informing model physical sizes. The right side represents observations of model instances in 3D scenes with unobserved latent scene scales. The two sides interact through the assumption that the product of model instance scales r_i and an overall scene scale r_s should take models from virtual dimensions v to physical dimensions p, as does the model scale r_m. Formally: $p = vr_m = vr_ir_s$ or $r_m = r_ir_s$.

We will treat the conditional probability of a given model having virtual size v: $P(v|p, r_m) = \mathbb{1}(v = p/r_m)$ as a deterministic relation involving the physical size of the represented object p and the latent scale of the model r_m. Since we expect to see variation in the observed model instance scales, we will model them as being drawn from a log-normal distribution: $P(\log(r_i)|r_m, r_s) \sim \mathcal{N}(\log(r_m) - \log(r_s), \sigma^2)$. As r_i is a ratio quantity, the variation in its distribution can be viewed as a product of independent random variables. Therefore r_i is likely to follow a log normal distribution. Working in log space also ensures that r_i is always positive and accommodates a wide range of scales across orders of magnitude. We will similarly treat r_m and r_s as log-normally distributed: $P(\log(r_m)) \sim \mathcal{N}(\theta_m, \sigma_m)$ and $P(\log(r_s)) \sim \mathcal{N}(\theta_s, \sigma_s)$. Although here we use relatively simple distributions over our random variables, a probabilistic framework allows for principled future extensions. We discuss improvements to the model, such as using multi-modal distributions or priors over the model scales in Section 7.1.

For the physical and virtual units we use the 3D model's bounding box diagonal. The diagonal is relatively stable with respect to axis aligned rotations in model alignment, which constitute the majority of cases of alignment inconsistency in our model datasets (see Section 4.1). A more advanced approach might use multivariate Gaussians for the distribution of physical and virtual dimensions, capturing additional information about the variability along each dimension.

We will first look at the right side of our probabilistic model where we only deal with observations of model instances in scenes, without a notion of model categories (Section 5.1). We will show that just using scene information we can get fairly accurate sizes for models observed in scenes. However, we cannot cover models that are not observed in any scenes.

To address this shortcoming, we will use categories and category size priors. This corresponds to the left side of our probabilistic model which we will first look at in isolation in Section 5.2. We will describe how we collect and use priors for physical sizes p and then how we determine model scales r_m for given models with virtual sizes v. Then we will connect these two approaches by using the entirety of our probabilistic model (Section 5.3).

Different information sources impact the extent to which our approach can cover a model dataset, as illustrated in Figure 3. Using scenes alone, we can only expect to predict sizes for models occurring in scenes (red). With just category size priors, we can only predict sizes for models of categories with priors (blue). By combining *scenes* and *categories*, we expand our coverage of sizeable models to include all models that have been categorized (light blue). However, 3D models that are not categorized and do not occur in any scenes will still be unsizeable (light gray). So far, we assumed that the category c is observed. When there are no manually assigned categories, c can be treated as a latent variable. In Section 5.4 we present a categorization algorithm to infer a value for c. Before the technical details, we first discuss our information sources.

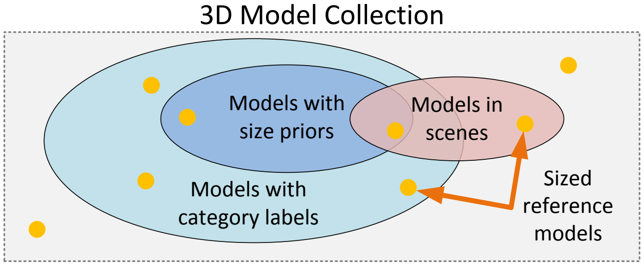
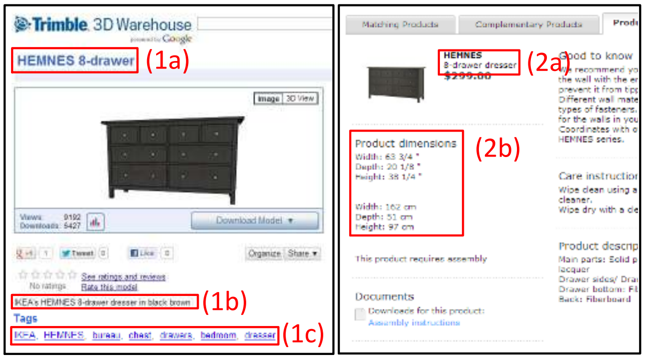


Figure 3: *Representation of a collection of 3D models and the partitioning of the models into sets based on available information.*



HEMNES 8-drawer

Views 9192
Downloads 5427

See ratings and reviews
No ratings
Rate this model

IKEA's HEMNES 8-drawer dresser in black brown

Tags
IKEA, HEMNES, bureau, chest, drawers, bedroom, dresser

HEMNES
8-drawer dresser
$299.00

Good to know
We recommend yo
the wall with the er
prevent it from tipp
Different wall mate
types of fasteners.
for the walls in you
Coordinates with o
HEMNES series.

Product dimensions
Width: 63 3/4 "
Depth: 20 1/8 "
Height: 38 1/4 "

Width: 162 cm
Depth: 51 cm
Height: 97 cm

Care instruction
Wipe clean using a
cleaner.
Wipe dry with a cle

This product requires assembly

Product descrip
Main parts: Solid p
lacquer
Drawer sides/ Drav
Drawer bottom: Fi
Back: Fiberboard

Documents
Downloads for this product:
Assembly instructions

Figure 4: *Left: model webpage, right: furniture page. 1a: model name, b: description, c: tags; 2a: furniture name, b: dimensions.*

4 Information Sources

The availability of data to use for training each part of our probabilistic model is an important consideration. Our primary input is a collection of 3D models (Section 4.1). We also use a dataset of 3D scenes created with a subset of these models for retrieving relative size observations (Section 4.2). Our absolute size information for learning category size priors comes from online furniture catalogue websites (Section 4.3). Finally, we use a small set of reference 3D models with known absolute sizes both as an information source and for evaluating our results (Section 6). Figure 4 gives examples of our information sources. In the following sections we examine each information source and present evidence that further motivates the need for consistent scaling of 3D models.

4.1 3D Model Databases

We use two 3D model datasets retrieved from large public 3D model repositories. One consists of 12490 models collected from the Google 3D Warehouse (now Trimble) by crawling for a variety of terms relating to indoor objects.[1] We refer to this dataset as 3DW. The second dataset is a complete crawl of the Archive3D.net (AR3D) repository which consists of 30062 models.[2]

After collecting these datasets, we semi-automatically annotated approximately 70% of all models with category labels. We used model tags to set these labels, and verified and augmented them manually. We defined a category hierarchy with 270 categories, of which 29 are parent categories with a total of 104 subcategories and 137 are childless parent categories. We chose categories so that a subset can be associated with size priors (Section 4.3).

We observe that most models we collected are consistently aligned. Though alignment in the horizontal plane varied, upright orientation was consistent for most objects and variations were primarily

[1] 3DW crawled during February 2012

[2] AR3D crawled during September 2012

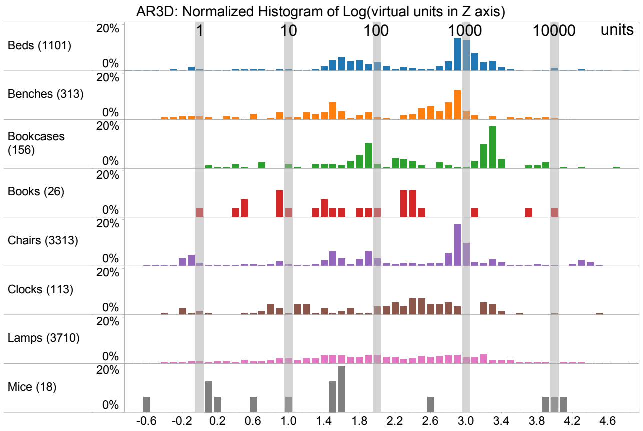


Figure 5: *Normalized histograms of the logarithms of vertical sizes for some categories of models. Model virtual units are used directly. The horizontal axis is logarithmic so the distributions are over orders of magnitude in linear units (top).*

due to 90 degree rotations in the horizontal plane, to which the diagonal is mostly stable. For example, 415 out of 419 tables in 3DW and all 1101 beds in AR3D have their canonical Z coordinate axis aligned with the upwards direction. To investigate the effect of misalignment on size estimation we perform an experiment where the 3D pose of models is randomized by choosing Euler angles randomly. We find that the performance of our size estimation algorithm is not impacted significantly—overall size prediction accuracy decreased by 2.1% relative to results presented in Section 6.2.

In addition to the geometry of each model, we also retain the text occurring in the webpage from which the model is downloaded. Textual information can be critical for providing additional knowledge about the model. Previous research has noted that the text in such online 3D repositories is often sparse or inaccurate [Goldfeder and Allen 2008]. While the quality of text can be lacking, it is still a highly discriminative information source that we use for categorizing model in Section 5.4.

Figure 5 shows normalized histograms of the vertical heights for several categories of models. Many categories such as books, clocks and computer mice have broad height distributions over several orders of magnitude indicating inconsistent scales. In the presence of broad and noisy distributions such as these, guessing a single "correct" scale is not a practical approach. Furthermore, though guessing scales such as "1 unit = 1 inch" to put particular models into physical units can work well for unimodal distributions, we cannot a priori confirm that these guesses are correct without external information. We will focus on the case where the model database is of unknown quality and make no assumptions about the nature of existing model scales. This allows us to handle 3D model datasets with arbitrary model scale distributions.

4.2 3D Scenes

If 3D scenes are available, we can size instances of models observed in a scene with objects of known size. Unfortunately, there are few 3D scenes in online 3D model repositories, which means that the vast majority of 3D models are observed out of context. For example, the number of 3D scenes on Google 3D Warehouse is miniscule (thousands) compared to the total number of models (millions).

To collect our model relative size observations we use a recently published 3D scene dataset of 133 small indoor scenes created with 1723 3D Warehouse models [Fisher et al. 2012]. Since this dataset primarily contains small indoor scenes, we added an additional set of 18 larger outdoor and indoor open space scenes, as well as 16 room interiors using the same scene design tool and model dataset, kindly provided by the authors. Furthermore, we recruited 20 participants and instructed each to create two scenes: one starting from an empty room with a bookcase and one from an empty room with a desk. In total, we have a 207 scene dataset, comprising scenes from previous work as well as the 74 scenes we created.

During the above scene construction experiments, we log the UI interactions of each participant and compute an estimate for the time spent rescaling models. This estimate counted the contiguous time between consecutive scale actions with less than 0.5s between them (actions were 1.05 or 0.95 uniform multiplicative rescalings). Total scaling time averaged over all participants and scenes was approximately 10% of total scene creation time. Comparing this to 9% for rotation and 25% for model search and retrieval, we see that the burden of rescaling models during 3D scene design is significant.

4.3 Category Size Priors

As described in Section 4.1, we defined a set of categories over our model datasets, for which we would now like to obtain size priors. We first investigate the usability of the 2D image-based approaches. We compared against both the results of Lalonde et al. [2007] from a set of 13000 LabelMe object instances (149 categories), as well as a 49 category subset provided by the authors of LabelMe3D [2009]. Overall, the overlap with the categories of our 3D models and ground truth data was limited (a dozen of mostly indoor categories such as chairs, tables, books and benches). Furthermore, we observed a consistent upward bias in the height priors derived from images, likely due to over-estimation of physical heights in approximating 3D objects as 2D planes perpendicular to the ground. The bias was particularly severe for objects typically not on the ground (for example, books were inferred to have a mean height of 2.7 m and cups 1.0 m). We suspect this is due to these methods using perspective back-projection with estimated camera parameters and assuming that all objects lie on the ground plane. This assumption is reasonable for outdoor scenes which were the focus of that work, but is unacceptably restrictive in indoor settings. For example, in the 3D scene dataset of Fisher et al. [2012], only 27.1% of objects are supported by the ground.

Since the height estimates from 2D images are not usable for our problem, we extract size priors corresponding to a subset of our categories by aggregating and processing textual descriptions from online furniture websites. We choose this source of information because furniture dimensions are well specified, easily accessible and reliable. Furthermore, our model datasets contain many indoor objects with furniture comprising approximately half of all models.

We scrape the websites of two online furniture retailers.[3] We extract the dimension DOM elements from the HTML pages of each furniture item and convert measurements to meters. For each furniture item we automatically map the manufacturer's categorization—also available in the HTML page—to our category hierarchy. We collect a total of 3099 furniture items in 55 categories. We then aggregate all dimensions per category and treat them as samples to fit a Gaussian prior for the probability distribution of physical sizes p: $P(p|c)$ of the particular category c. Thus, we augment a subset of the models in our database with category size priors from trustworthy external information. Figure 6 illustrates size priors for some of the furniture categories we collected. We provide this data, along with estimated size priors.

[3] `http://www.furniture.com` and `www.ashleyfurniture.com`

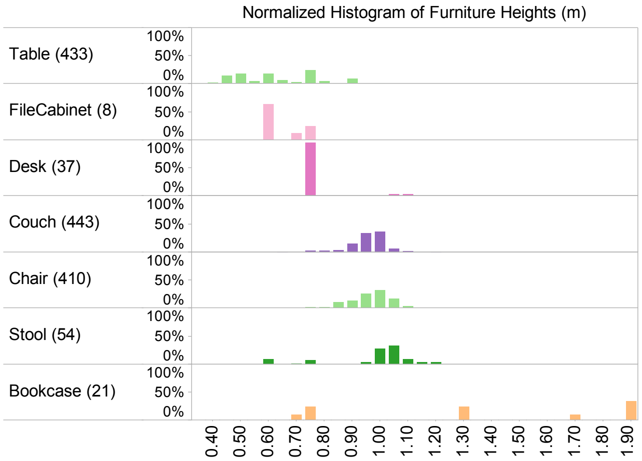


Figure 6: *Data collected from furniture product webpages. The plot shows normalized histograms of vertical height (* m*) distributions for a few categories of furniture. We use this raw data to create a set of size priors for 55 furniture and indoor item categories.*

5 Algorithm

We present our method for estimating model scales using these two information sources independently and in combination. Each method is evaluated independently and also compared against human judgments and ground truth data in Section 6. We show that the method combining information sources outperforms the simpler independent algorithms.

5.1 Leveraging Scene Information

When users construct scenes, they implicitly provide us with judgments about the sizes of objects relative to other objects in the same scene. Specializing our graphical model by omitting categories leads to a linear least squares optimization computation which outputs estimates for the model scales.

We ignore the left hand portion of the graphical model (Figure 2) and focus on the right hand side where we only have relationships between model scales r_m, scene scales r_s and model instance scales r_i. For any instance i of a particular model m occurring in a scene s, we have the relationship $\frac{r_m}{r_s} = r_i$. As discussed in Section 3, we express these scale variables logarithmically. Thus we express the relationship in log space with Gaussian noise

$$\log r_m - \log r_s = \log r_i + \epsilon, \text{ where } \epsilon \sim \mathcal{N}(0, \sigma)$$

By converting each observed instance of an object in a scene into such an equation, we arrive at a system of linear equations with unknowns $\log r_m$ and $\log r_s$. Since we have assumed Gaussian noise, the maximum likelihood estimate of this model can be found via linear least squares, which can be solved easily with a general sparse linear system solver.

Because there are two unknown variables for each constraint, in general the above system can be under-determined, in which case we must disambiguate between potential solutions by grounding one of the variables to a known value. This corresponds to specifying known model scales for at least one model per connected component in the graph of scene observations, thus removing the extra degree of freedom for the component. We use our reference models for this purpose. For each reference model, we remove the relevant variable r_m from the list of unknowns and substitute the known scale into every equation the variable appears.

5.2 Using Categories to Generalize

Categories generalize information from particular instances to groups of objects. They allow us to connect prior knowledge to 3D models in a principled manner. The approach we take here corresponds to the left part of the probabilistic model in Figure 2. If we assume that the physical size p for models of a given category c is drawn from a normal distribution, i.e. $P(p|c) \sim \mathcal{N}(\mu_c, \sigma_c)$, the maximum likelihood estimate for p is simply the mean μ_c for that particular category. As we will show in Section 6.2, this method allows for good coverage of the 3D model dataset and is very effective when combined with scene information. By itself, it gives plausible approximate sizes which are limited in their precision.

We note that, though categorization of 3D models is a separate research problem which is beyond our direct focus, the effort required to manually categorize large 3D model databases is significant. We therefore present an automatic categorization algorithm for 3D models in Section 5.4.

5.3 Combining Scenes and Categories

What can we do for objects neither covered by our background knowledge nor covered by scenes directly? A few options are available. We can expand our categorical priors, create more scenes, or propagate knowledge using both category priors and scenes. We'll focus on the latter approach.

We integrate information from both relative size observations and category priors by viewing the model scales and scene scales as latent variables. This probabilistic approach allows us to reconcile inconsistencies or incompatibilities in the data in a natural way. We are interested in the probability distribution of the model scales r_m, so we will treat the scene scales r_s as parameters in our model and use an iterative algorithm based on Expectation Maximization [Dempster et al. 1977] to find the maximum likelihood estimates (MLE) for the scene scales and the category size means. Once we have optimized the scene scales, we can use our model to obtain either a probability distribution for the model scales $P(r_m|c, v, r_i; r_s)$ or to pick the most likely value for a given model scale. Our algorithm is given as pseudocode in Algorithm 1. Note that under the assumption of Gaussian distributions, this algorithm simplifies to iterating between computing MLE for scene scales and category means, and MLE for model scales. However, the above algorithmic outline allows for more complex prior distributions to be incorporated by adjusting the update steps.

To initialize at a good starting point and avoid propagating noisy information, we keep track of which models, categories, and scenes we have scale estimates for, and use only them in each iteration. We start with known model scales r_m for the set of reference models $\mathcal{R}$, and known category size means μ_c for categories with collected size priors. These are then used to estimate scene scales r_s. Once we have some known scene scales, we compute new estimates for more model scales r_m. With these new model scales we can in turn compute updated estimates for more scene scales r_s and category size means μ_c. Through this iterative approach, we eventually cover all models reachable through either categories or scene observations. We terminate when no more new models are sized. For the results presented here we use 3 iterations. In general, the number of iterations is bounded by the length of chained model co-occurrence observations across the scene set (i.e. the diameter of the biggest component in the graph of connected co-occurring models).

This combined algorithm can be reduced to the approach of Section 5.1 by removing propagation of category information and not updating model scales using that information. It can also be reduced to the category size priors approach in Section 5.2 if we do not provide scene data.

Algorithm 1: EM for size propagation.

```
input  : set of 3D models $\mathcal{M}$= $\{m = (c, v)\}$
input  : set of scenes $\mathcal{S} = \{1 \ldots S\}$
input  : set of categories $\mathcal{C}$= $\{1 \ldots K\}$
input  : set of priors on physical size conditioned on category
         $\mathcal{P} = \{(c, \mu_c^0, \sigma_c^0)\}$
input  : set of reference models $\mathcal{R} = \{(m, r_m)\}$
input  : set of model observations in scenes $\mathcal{I} = \{(s, m, r_i)\}$
// Initialize mean of log of model scales $\theta_m$ and scene scales $\theta_s$ to {}
// Initialize $\theta_m$ for ref models
foreach $(m, r_m)$ in $\mathcal{R}$ do
    $\theta_m = \log(r_m)$
// Initialize category size mean $\mu_c$ to prior mean
foreach $(c, \mu_c^0, \sigma_c^0)$ in $\mathcal{P}$ do
    $\mu_c = \mu_c^0$
// repeat for T iterations until convergence:
for $t$ = $1$ to $T$ do
    // E-step: update MLE for non-ref model scales using estimated scene
       scales and category means
    foreach model $m$ in $\mathcal{M}$ - $\mathcal{R}$ do
        // Average over $N_{mi}$ = # of inst. scales for model $m$
        $\theta_m = \frac{\log(\mu_c)+\sum_i(\theta_s+\log(r_i))}{\mathbb{1}(\mu_c>0)+N_{mi}}$
    // M-step: update MLE for scene scales and category means using
       estimated model scales
    foreach scene $s$ in $\mathcal{S}$ do
        // Average over $N_{si}$ = # of inst. scales for scene $s$
        $\theta_s = \frac{\sum_i(\theta_m - \log(r_i))}{N_{si}}$
    foreach category $c$ in $\mathcal{C}$ do
        // Average physical size for category $c$
        $\mu_c = \frac{1}{N_c}\sum_m v_m \exp(\theta_m)$
output : estimated mean of log of model scales $\theta_m$
output : estimated mean of log of scene scales $\theta_s$
```

5.4 Automatic Categorization

Our method relies on categories to establish and propagate size priors. We have defined a manual categorization for our models, but to handle an open-world scenario we need to automatically categorize beyond a small set of pre-defined categories. There is much literature on retrieval and classification of 3D models using a variety of approaches [Min et al. 2004; Tangelder and Veltkamp 2008]. The problem of unsupervised 3D model categorization is a challenging one, and not our direct research focus but we present a sensible approach leveraging both text and geometry. To motivate our algorithm we first describe the semantic taxonomy we use, and then empirically compare the predictive performance of text and geometry features on our 3D model dataset.

Taxonomy Our algorithm maps each model to a node in the WordNet hierarchy [Miller 1995]. WordNet is a lexical database for the English language which groups English words into sets of synonyms known as synsets, roughly corresponding to a semantic category. It provides short, general descriptions of each synset and the synsets are arranged in a hierarchy containing hypernyms (wider categories) and hyponyms (narrower categories). Since we are interested in physical objects we consider only children of the "physical object" synset. We also filter out synsets corresponding to processes, locations and events. There are approximately 40 thousand physical object noun synsets that are valid prediction targets for our algorithm.

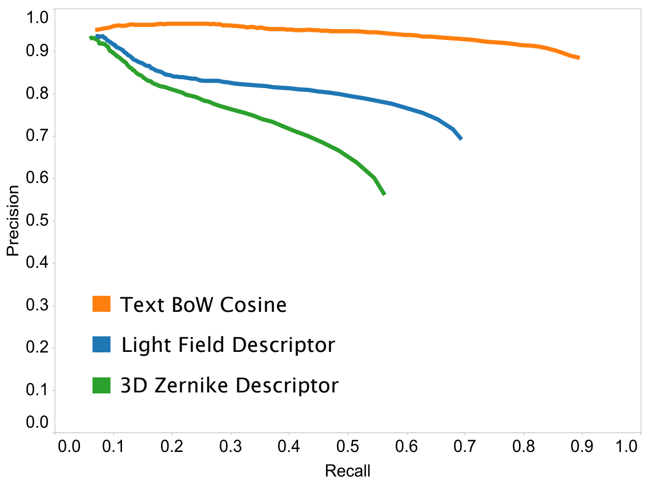


Figure 7: *Precision-Recall plot comparing 3D Zernike descriptors (green), light field descriptors (blue) and bag of words cosine similarity (orange) performance in predicting WordNet synsets for 30225 models, with 10-fold cross validation.*

Evaluation Setup For evaluating our categorization algorithm, we manually map the category hierarchy used by our model datasets to matching WordNet synsets. Of our 270 categories, 252 are mapped to a WordNet synset. The remaining 18 are not found in WordNet and consist mainly of recent electronic devices such as video game consoles and USB drives. With this mapping, we have synset correspondences for 28243 models (out of 30225 models with category labels) which we treat as our evaluation set.

Comparing Text and Geometry We use the above evaluation set to see how well we can propagate synset labels using text and geometry features. We base our approach on prior work dealing with propagation of text tags through geometrical similarity [Goldfeder and Allen 2008]. The method uses a distance-weighted nearest neighbor voting scheme, embedded in the space of the chosen geometrical descriptor. We experimented with both the 3D Zernike descriptors of Novotni and Klein [2003] as implemented by Goldfeder and Allen (128^3 voxelization, 20 moments resulting in 121 dimensions) and also with the light field descriptors (LFD) of Shen et al. [2003]. Though we chose simple and well-known methods there is much prior work in shape similarity measures that could be applied to this problem—a survey is provided by Tangelder and Veltkamp [2008]. We compare these geometry features against a bag of words cosine similarity measure which is standard in information retrieval [Manning et al. 2008]. We perform 10-fold cross validation on the above dataset and measure precision and recall to evaluate the predictive strength of each method (see Figure 7).

The bag of words cosine similarity feature using the model text performs much better than the geometry features—its average F1 score (harmonic mean of precision and recall) is 0.52 compared to 0.32 for the light field descriptors and 0.26 for the 3D Zernike descriptors. This is not surprising—since text search is still the primary retrieval method for 3D models, we can expect the text to be fairly indicative of the model category. Some categories such as microwave ovens, cardboard boxes and refrigerators are hard to distinguish using only geometry, but are easily disambiguated with text. Naturally, when there is no text we have to rely on geometry. The constraints of unsupervised categorization and these observations motivate the design of our algorithm. First, we use any available textual information to predict synset labels. Then, we propagate the results using geometrical similarity based on the light field descriptors to handle missing annotations and expand our coverage.

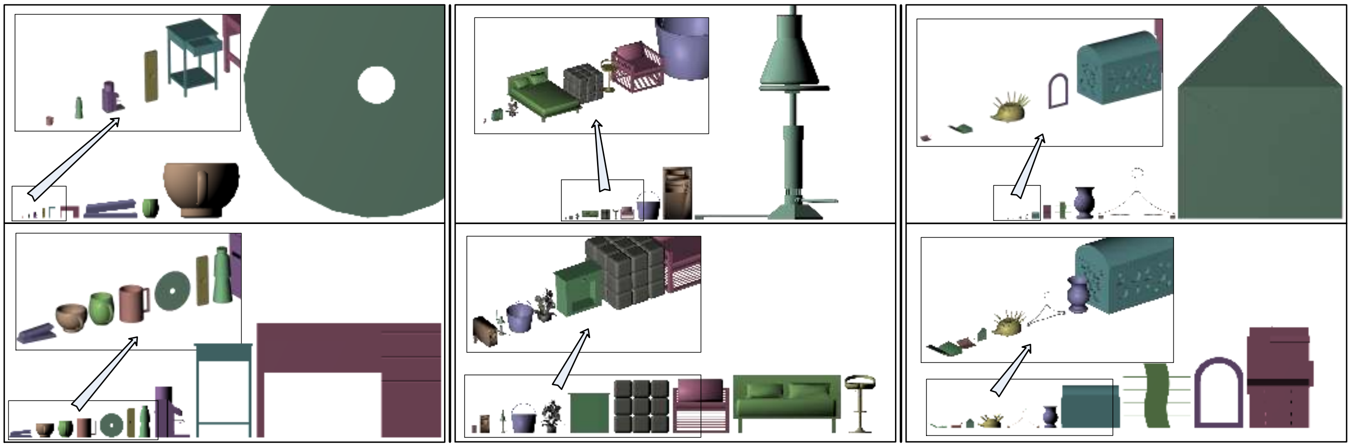


Figure 8: *Three sets of models selected through sorting by bounding box diagonal, binning into 10 bins and randomly sampling from the central 10% of each bin. Top: models in original scales. Bottom: Corrected scales using our* `Combo` *algorithm (unsized models in gray).*

Algorithm In the first stage, the names, tags and descriptions associated with each 3D model are preprocessed for tokenization, lemmatization and part of speech annotation using the Stanford CoreNLP pipeline [Toutanova et al. 2003]. Then we identify words in the model name that are most indicative of a matching synset. To do this we first match the entire name, then the longest possible phrase in the name, or extract and match the head word (the word that determines the semantic category of a phrase). If no match is found, we take nouns in the name and match them against synsets. Since each word can map to multiple synsets, we prefer synsets that are furniture and fixtures over synsets for body parts, people and media. Beyond this preference, we use the default ordering of synsets in WordNet, which follows sense frequency. To select among candidate synsets for different words, we take the TF-IDF (term frequency - inverse document frequency [Salton and Buckley 1988]) of words in the model text (t_m) and words associated with each synset (t_s). We then compute the cosine similarity between them $sim(m, s) = \frac{t_m \cdot t_s}{\|t_m\| \|t_s\|}$, and select the synset with highest cosine similarity. After this step, we have a set of 3D models that are partially corresponded to WordNet synsets. To handle cases where there was no textual information for a model or where no synset was matched with text, we use the propagation algorithm of Goldfeder and Allen [2008] with LFD features to select the highest probability synset from the 15 nearest neighbor models.

Results After the first stage, our algorithm automatically derives synsets for the 30225 models used for evaluation and we retrieve 21746 (72%) matches. 17251 (57%) of these matches are exact, 3191 (11%) belong to hyponyms of the target, and 1322 (4%) to hypernyms. The second stage then propagates synset predictions to unmatched models using light field descriptor similarity, matching a further 586 (2%) models correctly and bringing the output synset match accuracy to 74%.

The algorithm can be improved in a variety of ways. Shape similarity can be used to identify noisy text and help disambiguate between different word senses. Another potential improvement would be to use 3D scene context in categorization, whenever it is available [Fisher and Hanrahan 2010].

This categorization algorithm can predict any matching target WordNet synset, not just the categories with which we annotate our model datasets. We incorporate the output synsets of this algorithm into our framework by using them instead of manual categories. In Section 6.2, we present results using this automatic categorization.

6 Evaluation

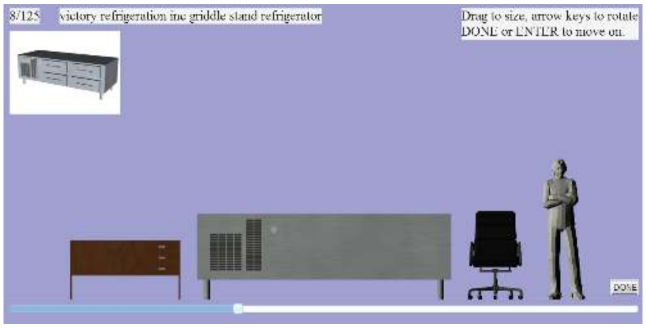
We first demonstrate our approach by using it to rescale randomly sampled models from the 3DW dataset and showing that corrected scales result in more plausible object sizes. Figure 8 shows three randomly sampled model sets, each spanning a range of sizes. We see that the rescaled versions are more plausible. The Rubik's cube in the middle remains at an implausible size because it is not covered by our algorithm (indicated in gray).

We evaluate size predictions from our algorithms against human size judgment and ground truth sizes. Section 6.1 describes how we collect size judgments from people. In Section 6.2 we compare human judgments against ground truth and our algorithmic predictions. We show that there is a large variation in human judgment and that our algorithm predicts sizes with more accuracy than people on average. By combining information from scenes and categories, we achieve higher accuracy and coverage than using each independently. Section 6.3 analyzes the impact of the number of available scenes, reference models and category priors. Finally, we show we can use automatic categorization to achieve improved coverage of our model database.

6.1 Human Size Judgment

We performed an experiment to collect human perceptions of 3D model size. The experiment was designed to require relative size judgments against reference objects to avoid having people perform a numerical value recall task. To create our evaluation model dataset, we first sampled from the combined 3DW and AR3D dataset to obtain 105 models. A third (35) were uniformly sampled from models observed in scenes, a third from categories with size priors and a third having neither observations nor priors.

We used a simple online interface that randomly presented the participant with a model from the evaluation set initialized at a random size (Figure 9). To the left and right of the focus model were two reference models: one of smaller height and one of larger height. As an additional reference, we included the figure of a person to the far right. The set of reference objects consisted of 6 models that we manually sized: a finger ring, a soda can, a CPU case with monitor and keyboard, a study desk, an office chair, and the person. The models were selected from categories with narrowly defined typical sizes and such that they cover a range of sizes. As the participant resized the focus model, the side models were automatically reselected to bracket the focus model in height and the view zoomed in for small objects. We recruited 20 participants (10 female) from the



8/125 victory refrigeration inc griddle stand refrigerator

Drag to size, arrow keys to rotate
DONE or ENTER to move on.

Figure 9: *UI for collecting human size judgments. Participants are presented with a random model in the center. On the sides are reference models to facilitate relative size judgments. Participants drag to rescale the central model. The reference models are automatically reselected to bracket the central model in height.*

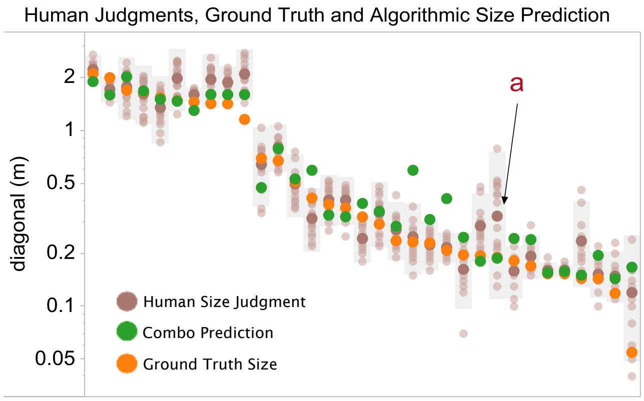


Figure 10: *Comparison of size responses by people (mean response is darker), ground truth sizes and size predictions by* `Combo`*. The vertical axis plots the 3D model bounding box diagonal in meters and is logarithmic. Each column represents one of 33 models in the evaluation set and columns are sorted by mean ground truth size.*

computer science department of a large university. Individual size predictions were aggregated and used to determine the acceptable ranges of size for each model.

We also manually collected ground truth absolute sizes for a total of 100 models, of which 33 are in this evaluation set. We do this by matching the 3D models to physical objects described in online product catalogues, or other known standardized sizes (such as DVDs, batteries and soda cans). We use the 33 models in the evaluation set for comparing human judgments against ground truth, and the remaining 67 as known size reference models for our algorithm. We provide both the human judgments for our evaluation set and the ground truth annotations as part of our supplemental materials.

Figure 10 shows the human judgments along with ground truth sizes and algorithmic predictions. We observe that human size judgments exhibit significant variation. The standard deviation compared to the ground truth data was 0.267 m. We normalize standard deviation against mean size for each model to get a relative standard deviation of 20%. This implies that human judgment exhibits 20% relative error with respect to a given absolute size. This result confirms previous research that shows human size judgments have significant variation, and can be influenced by context and familiarity [Fredebon 1992]. Results in the following section show that our algorithm can be more accurate than human size judgment.

Predictor	Human	InchU	InchP	`SS`	`CP`	`Combo`	`ComboWN`
RMSE	0.241	274.0	170.0	0.126	0.257	0.167	0.284
sized	33	33	33	22	9	33	32

Table 1: *Root-mean-square error in meters against ground truth object sizes for: mean human judgments, naïve guessing of inch scales on unperturbed and perturbed model evaluation sets, and the predictions from each of our methods. The* sized *row gives the number of models for which the method had a prediction.*

6.2 Comparing Humans and Algorithms

We evaluate our algorithmic size predictions against those collected from people. Since we do not assume a known distribution of scales, we randomly perturb the original model scales in order to avoid bias in the evaluation set models.

Using our 33 model ground truth set as the baseline, we compare the error of different prediction methods for object scales. Table 1 reports the root-mean-square error (RMSE) in meters across all models for which each method can predict sizes. For brevity, we refer to our algorithmic methods as follows: `SS` for scene scales, `CP` for category size priors, `Combo` for combined with manual categorization and `ComboWN` for combined with WordNet categorization. To compare with the accuracy of human judgment, we use the mean human-estimated model size for each object. We also compare against a default inch scale on the original (InchU), and perturbed 3D models (InchP).

From Table 1 we observe that using scene scales (`SS`) has the least error with respect to ground truth. While not as accurate as `SS`, using mean category size (`CP`) has comparable error to human judgment. By combining scene scales and categories, `Combo` can predict sizes for more models but has a higher error than `SS`. Overall, `Combo` predicts ground truth sizes better than the mean human judgment. This is also reflected in Figure 10: some algorithmic predictions are closer to the ground truth than the human mean. For example, see (a) in the figure, where a pencil model was judged to be much larger than ground truth and our algorithmic prediction. Using a naïve approach of guessing inches, the resulting RMSE for both perturbed and original model scales is much higher than any other approach, indicating the limitations of guessing a single scale.

We now compare our algorithmic predictions directly against human judgments. We count an automatically predicted size as correct with respect to human judgment if it is within 2 standard deviations of the mean size provided by people. Assuming human judgment is normally distributed around a correct size, this corresponds to the predicted size being statistically indistinguishable from human judgment at a 95% confidence level. Figure 11b summarizes the performance of our algorithm on the evaluation set. Each method is evaluated on the subset of models for which it can be applied (evaluation subset accuracy), as well as the entire evaluation set (evaluation set accuracy). The former gives us a sense of how well the method performs on models it can cover while the latter gives an indication of the overall performance of the method taking its coverage into account. Taking the evaluation subset accuracies, we can predict how well our algorithm will do on our entire model dataset based on the coverage of each method as shown in Figure 11c.

Overall, we note that using scene scales (`SS`) gives the best evaluation subset accuracy (88%), with other methods not far behind. The high evaluation subset accuracy for all methods indicates that any of these methods can achieve good results against human judgment for the models they can cover. However, despite `SS`'s high evaluation subset accuracy, it actually has the worst projected model dataset accuracy (6%) due to its limited coverage (6%). In contrast, using category size priors (`CP`) gives much higher coverage (52%) and thus better overall accuracy (42%) despite lower evaluation subset accuracy. By combining the two in Combo we increase the coverage significantly (72%) and obtain higher overall accuracy (58%), showing that we can effectively propagate size information using both scenes and categories.

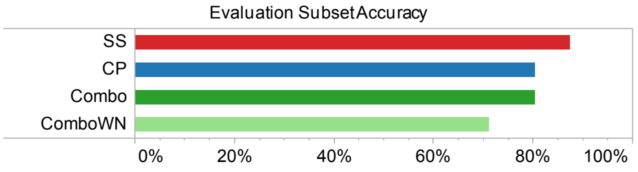
An important factor in performance is categorization. While ComboWN has slightly lower evaluation set accuracy than Combo, it improves overall coverage (84%) and accuracy (60%) by automatically categorizing models. The coverage of ComboWN is not confined by missing category labels and extends into previously uncategorized models (gray area).

6.3 Impact of Reference Models, Scenes, Size Priors

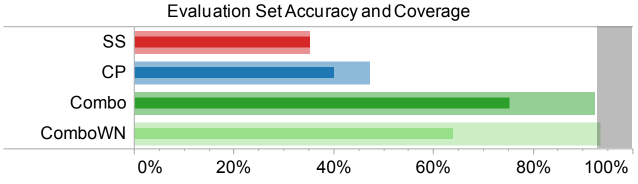
We analyze the impact of varying the number of available 3D scenes, the number of sized reference models and the number of category size priors. This gives us a sense of how the performance of our approach changes under situations with different amounts of input data. Analyzing the behavior of our algorithms under such variation is important because the availability of each information source varies across different datasets.

We test how the coverage over a combined (3DW+AR3D) 42327 model dataset and the overall accuracy on the 105 model evaluation set change as we increase each resource. In each case, we select the most informative resource first. For scenes, we select the ones with most model instances first. The intuition is that scenes with the most models give us the highest number of model observations. For reference models, we select models in descending order of the number of observations in different scenes. Lastly, for category size priors, we select the ones with largest model dataset coverage first.

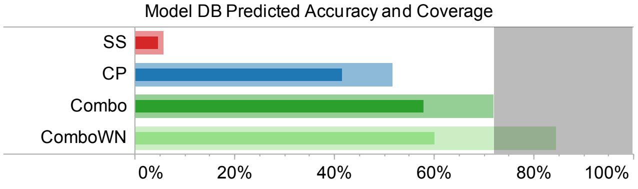
Figure 12 plots the accuracy on the 105 model evaluation set across these dimensions of variation—coverage is not shown as it largely follows the same trend. Based on our experiments, coverage and accuracy on the evaluation set is primarily determined by the number of scenes and the categorization scheme. With more scenes both the coverage and accuracy increase. In contrast, increasing the number of reference models or the number of category size priors does not impact coverage or accuracy significantly.



(a) *Evaluation subset accuracy, defined as percentage of correct models for subset of models to which method is applicable.*



(b) *Overall accuracy (inside bar) and coverage (outside bar) on the evaluation set.*



(c) *Predicted accuracy and actual coverage over combined dataset (3DW+AR3D). Predicted accuracy is extrapolated from partial accuracy and coverage.*

Figure 11: *Performance of our methods on the 105 model evaluation set and the combined dataset (3DW+AR3D). Gray represents models with neither manual categories nor observations in scenes.*

We now separate out the contribution of each component of the Combo method. From the middle column of Figure 12, we see the benefit of just adding category label information to SS. We introduce a variant of Combo that does not have access to category size priors ($\text{Combo}_{\texttt{NoPriors}}$). However, $\text{Combo}_{\texttt{NoPriors}}$ does take advantage of category labels and is able to propagate size information from scenes into a larger set of models with matching categories, achieving higher accuracy—see (a). There is an additional gain in accuracy from adding the category size priors—see (b)—which is purely due to more accurate seed information.

To study the impact of reference models, we use a variant of Combo which has no reference models: $\text{Combo}_{\texttt{NoRefs}}$. Just by adding scenes without any reference models, we surpass the performance of CP substantially—see (c). Again, the contribution of the reference models—see (d)—is due to more accurate seed information.

6.4 Summary of Results

We observed that 3D scenes are a good source of size information and, along with known size reference models, result in the most accurate size predictions. An approach using them is ideal when available scenes cover a large proportion of a model dataset. However, the number of scenes is typically small compared to models so the coverage of this approach is limited. Using categories we can propagate size information to a much larger set of models and generalize observations from scenes. However, relying on categories alone results in less accurate predictions.

We showed that by combining categories and scenes we obtain high model dataset coverage (72% of 42327 models) and size prediction accuracy (80%), as well as prediction error lower than human judgment. Furthermore, we showed that we can bypass manual categorization via an automatic categorization scheme that improves coverage to 84% (ComboWN). In this way, we address the *open-world problem* of continuously expanding model datasets with unknown or untrustworthy object categories and size information.

7 Discussion and Conclusion

7.1 Limitations and Extensions

Our approach has several limitations that suggest avenues for future work. Firstly, we explored only unimodal Gaussian priors for category sizes. To handle categories with latent subcategory structure, we might use Gaussian mixtures as priors for category sizes. This would for example handle bookcases with varying numbers of shelves. Furthermore, for models that can be interpreted as physical objects of different sizes (such as toy airplanes vs real airplanes) we might incorporate the context of the model instance to inform sampling of the size prior. This can be used to facilitate scene creation by automatically suggesting appropriate sizes, depending on the context. If we place an airplane model inside a hangar, it should be much larger than when placed upon a desk.

Another avenue for future work is to use external knowledge for a more informed prior on the expected model scales. This can take the form of a mixture of Gaussians on scales, corresponding to standard units such as inches, centimeters, millimeters and meters.

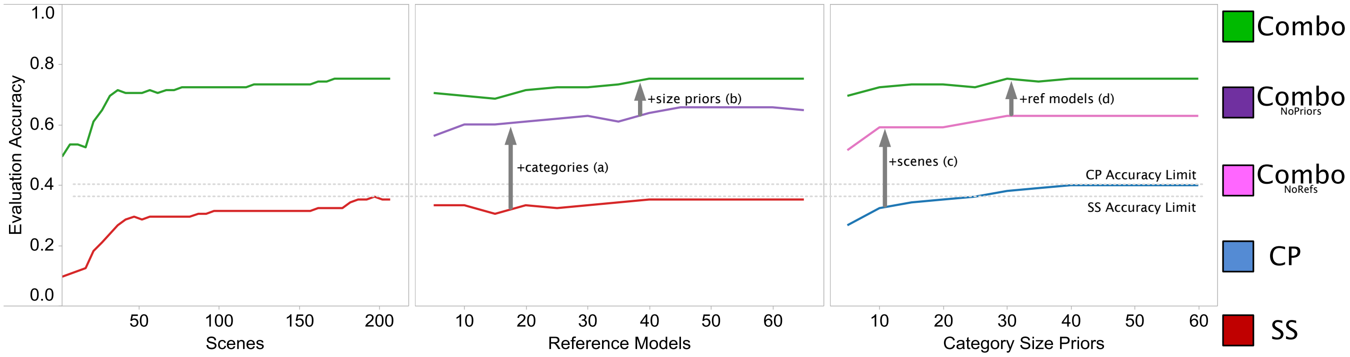


Figure 12: *Plots of the overall accuracy on the 105 model evaluation set. Left column varies number of scenes available to each method. Middle column varies number of sized reference models. Right column varies number of category size priors.*

We focused on an algorithmic approach to predicting 3D model scales. An alternative is to use crowdsourcing platforms such as Amazon's Mechanical Turk to collect size predictions from people. However, as we have seen in our evaluation, care should be taken since human size judgments can be unreliable. Typically, crowdsourcing requires verification and sanity checking so our method can be complementary to this approach by suggesting good starting scales for confirmation, or validating human input and flagging dubiously scaled models.

7.2 Conclusion

In this paper, we addressed the problem of determining model scales to give plausible real-world sizes to collections of 3D models. We presented an approach combining information from 3D scenes, category size priors and known size reference models. Our approach uses scenes and categories to generalize beyond observed instances and propagate size information to large collections of 3D models. We showed that our approach obtains favorable results evaluated against both human judgment and ground truth data. The probabilistic framework we have presented when formalizing this problem can be used to extend the approach and to address limitations. We provide all collected size prior data, 3D scene datasets, ground truth size annotations and predicted 3D model sizes for the benefit of the research community. We hope our work will inspire others to investigate applications utilizing previously unavailable size information for large collections of 3D models.

Such size data can be used to improve high level 3D scene synthesis algorithms and interactive systems. Model databases augmented with clean size metadata are of great value to the wider research community and 3D content creators. Probability distributions over sizes of categories of objects can be used as input to classification and object recognition systems. Novel 3D model search interfaces can leverage size data to allow retrieval and navigation with size ranges or size words. A knowledge base of 3D models with physical sizes can enable powerful forms of inference such as predicting the affordance of graspability for hand-sized objects with shapes similar to cups. We believe that reliable size data for 3D models can have far reaching implications in computer graphics and many other fields.

Acknowledgements

Support for this research was provided by the Stanford Graduate Fellowship fund and Intel (ISTC-VC). We would like to thank all the 3D warehouse users who created the models found in our scene database.

References

Barnes, M., and Levy Finch, E. 2008. Collada: Digital asset schema release 1.5.0 specification. *Khronos Group, SCEI*.

Davidov, D., and Rappoport, A. 2010. Extraction and approximation of numerical attributes from the web. In *In Proc. of 48th ACL*, ACL.

Dempster, A., Laird, N., and Rubin, D. 1977. Maximum likelihood from incomplete data via the EM algorithm. *JRSS*.

Fisher, M., and Hanrahan, P. 2010. Context-based search for 3D models. *ACM TOG*.

Fisher, M., Ritchie, D., Savva, M., Funkhouser, T., and Hanrahan, P. 2012. Example-based synthesis of 3D object arrangements. *ACM TOG*.

Fredebon, J. 1992. The role of instructions and familiar size in absolute judgments of size and distance. *Attention, Perception, & Psychophysics*.

Fritz, M., Saenko, K., and Darrell, T. 2010. Size matters: Metric visual search constraints from monocular metadata. *NIPS*.

Goldfeder, C., and Allen, P. 2008. Autotagging to improve text search for 3D models. In *8th ACM/IEEE-CS conference on DL*, ACM.

Hoiem, D., Efros, A. A., and Hebert, M. 2006. Putting objects in perspective. In *CVPR*, vol. 2, IEEE, 2137–2144.

Kraevoy, V., Sheffer, A., Shamir, A., and Cohen-Or, D. 2008. Non-homogeneous resizing of complex models. In *TOG*, ACM.

Lalonde, J.-F., Hoiem, D., Efros, A. A., Rother, C., Winn, J., and Criminisi, A. 2007. Photo clip art. In *TOG*, vol. 26, ACM, 3.

Manning, C. D., Raghavan, P., and Schütze, H. 2008. *Introduction to information retrieval*, vol. 1. Cambridge University Press Cambridge.

Miller, G. 1995. WordNet: a lexical database for english. *CACM*.

Min, P., Kazhdan, M., and Funkhouser, T. 2004. A comparison of text and shape matching for retrieval of online 3D models. *Research and Advanced Technology for Digital Libraries*.

Novotni, M., and Klein, R. 2003. 3D zernike descriptors for content based shape retrieval. In *8th ACM symposium on solid modeling and applications*, ACM.

Russell, B., and Torralba, A. 2009. Building a database of 3D scenes from user annotations. In *CVPR 2009*, IEEE.

Salton, G., and Buckley, C. 1988. Term-weighting approaches in automatic text retrieval. *Information processing & management*.

Satkin, S., Lin, J., and Hebert, M. 2012. Data-driven scene understanding from 3D models. *BMVC*.

Shao, T., Xu, W., Zhou, K., Wang, J., Li, D., and Guo, B. 2012. An interactive approach to semantic modeling of indoor scenes with an RGBD camera. *ACM TOG*.

Shen, Y.-T., Chen, D.-Y., Tian, X.-P., and Ouhyoung, M. 2003. 3D model search engine based on lightfield descriptors. *Eurographics Interactive Demos*.

Tangelder, J. W., and Veltkamp, R. C. 2008. A survey of content based 3D shape retrieval methods. *Multimedia tools and applications*.

Toutanova, K., Klein, D., Manning, C., and Singer, Y. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. *NAACL*.

Wang, K., and Zhang, C. 2009. Content-aware model resizing based on surface deformation. *Computers & Graphics*.

Wohlkinger, W., Aldoma, A., Rusu, R., and Vincze, M. 2012. 3DNet: Large-scale object class recognition from CAD models. *ICRA*.

Zia, M., Stark, M., Schiele, B., and Schindler, K. 2011. Revisiting 3D geometric models for accurate object shape and pose. *ICCV*.