The Interdisciplinary Center, Herzlia
Efi Arazi School of Computer Science

Filling Your Shelves: Synthesizing Diverse Style-Preserving Artifact Arrangements

M.Sc. Dissertation

Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Science (M.Sc.) Research Track in Computer Science

Submitted by Lucas Majerowicz

Under the supervision of Prof. Ariel Shamir

December 2014

Acknowledgements

I would like to thank my advisor Prof. Ariel Shamir for his invaluable feedback, guidance and insight. His strong work ethic, passion for computer graphics and continued demonstrative generation of solid research is inspiring. I would also like to express my gratitude to Prof. Alla Sheffer and Prof. Holger H. Hoos for their contribution and expertise - I could not have completed this work without their help. Finally, I'd like to thank my partner Josie, for her constant support and faith in me.

Abstract

Our homes and workspaces are filled with collections of dozens of artifacts laid out on surfaces such as shelves, counters, and mantles. The content and layout of these arrangements reflect both context, e.g. kitchen or living room, and style, e.g. neat or messy. Manually assembling such arrangements in virtual scenes is highly time consuming, especially when one needs to generate multiple diverse arrangements for numerous support surfaces and living spaces. We present a data-driven method especially designed for artifact arrangement which automatically populates empty surfaces with diverse believable arrangements of artifacts in a given style. The input to our method is an annotated photograph or a 3D model of an *exemplar* arrangement, that reflects the desired context and style. Our method leverages this exemplar to generate diverse arrangements reflecting the exemplar style for arbitrary furniture setups and layout dimensions. To simultaneously achieve *scalability*, *diversity* and *style preservation*, we define a valid solution space of arrangements that reflect the input style. We obtain solutions within this space using barrier functions and stochastic optimization.

Contents

List of Figures

Chapter 1

Introduction

"God is in the details" is a well known aphorism, often attributed to the architect Ludwig Mies van der Rohe (1886-1969). As 3D scenes used in movies, games and virtual environments are becoming larger and more complex, it is the attention to fine details that provides both realism and interest. Recent automatic procedures for the creation and arrangement of 3D content have made it possible to synthesize and furnish living spaces [1–6]. However, real life spaces - such as kitchens, libraries and shops - are full of smaller scale artifacts such as books on shelves, or cups and plates in cabinets. These details bring such environments to life (see Figure 1.1), and are essential for creating a sense of realism in virtual environments. With the sheer amount of such items in living spaces, assembling these arrangements manually is impractical and automatic assembly methods for small scale objects are needed.

What makes the layout of fine objects distinct from coarser-level arrangement problems is the variety of objects used and the diverse composition or arrangement styles. The choices of which items to place, where to place them and how to arrange them depend both on the *context* of the scene (e.g. a home kitchen or living room, a bar counter in a restaurant, or a display cabinet in a shop) and on the overall look of an arrangement of the arrangement (e.g. tidy or unorganized, packed or airy). While it is hard to define a given look, or a *style* as these can be amorphous properties, in our context we focus on a set of properties commonly considered in interior design when arranging objects on

Figure 1.1: We propose an automatic method to believably fill shelves and cabinets with objects reflecting the style of a single user-specified exemplar (left). Such arrangements can bring to life virtual environments that otherwise feel dull and lifeless (right).

shelves [7, 8]. For simplicity, we treat both the context and the overall look as part of one property set we refer to as *style*.

Arrangements generated at random (e.g. Figure 4.2,b) appear chaotic and fail to capture aesthetic and functional relations between the objects or their relationships with the environment. On the other hand, "typical" arrangements built from a set of canonical rules lack personality and cannot capture the variety in styles and content present in real-life arrangements (See section 5.2 for discussion and examples). Explicitly codifying each context and style combination, may be possible for furniture arrangements where variation is lower [3, 5], but is too tedious in our setup due to the number of objects involved and the style variety.

We introduce an approach for style-preserving artifact arrangements that leverages a *single* input exemplar, a photograph or a 3D scene, to learn the target style. We focus on generating arrangements of objects placed on horizontal support-surfaces, such as shelves and cabinets, which include the vast majority of arrangements in our everyday surroundings. By learning style from a single exemplar, we can support a wide range of styles without explicitly codifying the rules for each one (e.g. Figure 1.2).

1.1 Overview

Using a single exemplar to generate arrangements presents a number of challenges. We aim to populate surfaces in a believable and *scalable* manner, creating arrangements that

FIGURE 1.2: The same cabinet filled with arrangements reflecting different context and style based on different exemplars: a bar and two different kitchen cabinets.

can significantly vary in size and dimensions compared to the exemplar. To address scalability we design a set of size-invariant measures that capture deviations in style between a given arrangement and an exemplar. We use two types of style measures: object level terms that capture local arrangement properties, such as the percentage of instances of a particular object and the relative location of objects (Section 4.1), and global terms that compare high-level characteristics of the two arrangements, such as density and symmetry (Section 4.2).

The biggest challenge when using a single exemplar, is to allow for *diversity*, or variety, of outputs while capturing style. Such diversity lets artists exercise their personal preferences in selecting an arrangement to use. More significantly, it allows our method to effectively generate many arrangements with similar functionality and style without those appearing as cloned copies.

Given our collection of measures or distance terms, standard optimization methods, such as those used for furniture arrangement [4, 9] or individual object placement [6], would seek to find an arrangement that minimizes some weighted combination of these terms. Unfortunately, since we have a single exemplar which clearly satisfies all the characteristics learned from it, this approach converges to results that are very similar to the exemplar, even when starting from a randomly initialized arrangement (Figure 1.3 top). To generate a range of diverse results from a single exemplar we propose to use a *valid space* formulation. Instead of seeking the **best** solution we optimize toward a **valid** one, i.e., a solution that is within an acceptable range from the exemplar with respect to the various style measures we defined. Using this approach the solution is

FIGURE 1.3: Minimizing a combination of distance metric with respect to a single style exemplar (taken from Figure 3.1) using random initialization leads to low variability (top). Our valid space approach captures the style of the exemplar while producing diverse results (bottom).

brought close enough to the exemplar to capture its style, but not too close, to retain diversity (Figure 1.3 bottom) and thus supports the generation of many distinct valid solutions.

We represent the valid space via a set of inequality constraints defined using barrier functions. To obtain solutions within the valid space we begin with an empty arrangement and apply a stochastic search strategy aiming to bring the solution into this space by applying a randomized sequence of simple modification operations such as placement, swapping and removal of objects.

The differences in the problem setting, such as style and believability criteria, prevent a direct comparison of our method to related layout techniques (e.g. furniture). Instead we provide a comparison to methods that mimic the core features of those, but use the style measurements introduced below. Our method better preserves style and generates more believable results when compared to methods that do not leverage local exemplar properties. At the same time it generates significantly more diverse results than unbounded optimization which seeks to minimize the distance to the exemplar across a sum of similarity measures.

We present a tailored optimization approach based on a set of scale-invariant measures which support arrangement scalability combined with the use of barrier functions which define a valid solution-space. This space contains numerous solutions that follow the style of the exemplar allowing output diversity. We validate our approach by conducting a user study showing that our arrangements are considered believable, and that they manage to capture the style of a given input exemplar (Section 5). We further demonstrate our method's versatility by generating a variety of arrangements, for different contexts and with various styles, creating rich and believable results (Section 6).

Chapter 2

Previous Work

2.1 Synthesis and Layout

Automatic synthesis and arrangement of 3D scenes are very active research topics. Recent papers in this area discuss everything from the design of city layouts [10, 11], through architectural design of individual buildings [2, 12] and interior floor plans [1], to the layout of furniture [3–5] and context-based placement of objects [6, 9]. Our layout of fine-scale artifacts in a given style is a natural next step, made even more necessary by the sheer amount of such artifacts in a typical environment and the diversity in the style of their arrangement. While some ideas from coarser level methodologies can be adopted for our task, the problem setup is distinct enough to require not only different domain knowledge but different overall formulation. In large scale content generation such as buildings or cities, the main challenge is the generation of new geometric shapes. Our main focus is on arrangement of pre-existing shapes in a believable manner and style following the "open world" assumption, where objects need to be chosen and not only positioned.

Furniture layout methods generally focus on generation of stylish [3] or typical [4–6] layouts. They typically do not explicitly address scalability, a critical requirement in our setup (Figures 1.1,5.5), as typical rooms do not dramatically vary in terms of size or number of furniture pieces. Most methods use design rules or relationship information learned from multiple 3D examples, or a combination of both. Neither approach is suitable for capturing individual style properties. First, the amount of style and context variation in artifact arrangements is too large to codify via explicit rules (Figure 1.2). Second, style cannot be learned from a set of heterogeneous examples, as mixing many examples dilutes any specific style and building a separate large database for each style and context combination is impractical. Reducing the number of examples in systems such as [4, 6, 9] to capture a specific style is likely to dramatically limit the output diversity as these methods optimize toward results as-similar-as-possible to the exemplar. In our experiments with artifact arrangements such optimization leads to clone-like solutions (Figure 1.3,top). Our approach is based on the definition of valid-solution space. This successfully achieves diversity while capturing the style of a single example input (Figure 1.3, bottom). An explicit comparison of these methods to ours is impossible due to domain differences, But sections 5 and 6 compare our method to alternatives inspired by these approaches highlighting the improvement in style preservation and variability.

At the finest level, Ma et al. [13] propose an elegant modeling mechanism to treat piles of small objects as 3D textures. They assume the input consists of multiple repetitive elements and apply random organization of element patches to generate the outputs. In our case this would be equivalent to randomly arranging sub-groups of exemplar elements. Extending this approach using suitable rules could potentially capture object level relationships (Figure 6 (b)) but cannot account for high-level style properties, such as symmetry or functional grouping (Figure 6 (e)).

In [1] Merrel and colleagues presented a method for automatically generating building layouts based on high level requirements. Their method also utilizes stochastic optimization and design guidelines to synthesise new results. However, having a different goal from ours, their technique cannot generate many different style-preserving results all based on a single input exemplar.

In [4] Yu and colleagues presented a method for automatically generating indoor scenes realistically populated by furniture objects. Similar to our technique, they use design metrics along with stochastic optimization to generate new results. However, their algorithm needs to be given the set of furniture objects to arrange. In our case, since we aim at synthesizing arrangements of any size, our method also needs to be able to determine which objects and how many to include.

2.2 Style Preservation

Recently, several papers addressed the problem of capturing style in different contexts. For instance, Doersch et al. [14] find architectural elements of cities in photographs, and Xu et al. [15] separate content from style in 3D man-made objects. Such approaches rely on the existence of a set of examples (photos or objects) conveying the same style, while we aim to define style using arrangement characteristics drawn from a single example.

A number of recent works use domain-specific design principles to create feasible content [3, 4, 16, 17]. This trend is summarized nicely in Agrawala et al. [18], who note that, given a set of design rules and quantitative evaluation criteria, one can use procedural techniques and optimization to build automated design systems. Interior design literature, e.g. [19], proposes several criteria to evaluate the aesthetics and functionality of an arrangement. Instead of globally optimizing these criteria, our aim is to obtain outputs which are similar to the exemplar in terms of these criteria, e.g., if the input is unorganized, then the output should be unorganized as well. This approach preserves the style characteristics of an exemplar in the output. To reflect individual arrangement features we combine these global measures with object-level ones.

In [9] Fisher et al. presented a method for synthesising 3D objects from examples. Their system relies on a large database of scenes and user-given examples to construct probabilistic models, which are then sampled in order to generate new scenes. Applying a similar technique for synthesizing style-preserving arrangements would require several input exemplars with the same style, which are not easy to come across or generate.

In [23] Kalogerakis et al. presented a method for synthesizing new shapes based on existing shapes. They use a probabilistic model of shape structure that, after having being trained, can be used for synthesizing new plausible shapes. For this method to work, it needs to be given a large number of example shapes. The problem with using a similar method for the problem that we are trying to tackle is that 3D examples arrangements with a similar style are hard to come across. For this reason, we aim at synthesizing new arrangements based on a single input exemplar.

Chapter 3

Algorithm

3.1 Algorithm Input

The models used to populate our arrangements can come from any 3D database of household items. We normalized the sizes of all objects to be on the same scale. Each object is assigned a general *type*, e.g. a plate, a cup, or a decoration, and an individual *label*, e.g. CUP1, CUP2, PLATE1, PLATE2, or MICROWAVE. In the following discussions we use a coordinate system where x is the support surface width, y is the height, and z is the support surface depth (when viewed head-on).

To generate an exemplar arrangement we typically rely on photos of real arrangements with a desired style. This choice is motivated by the fact that 3D examples of arrangements in various styles are difficult to create or find, while images of real arrangements can be easily obtained. To extract the objects and their relationships from a photo, we use a simple 2D annotation tool (Figure 3.1). The user can load any unobstructed, front facing image of an arrangement. She marks the individual support surfaces in the image, then marks a bounding box around each object appearing in the image, and assigns a type to it. The user also specifies the front-to-back order of objects, if they appear behind each other. Each type of object appearing in the exemplar is associated with an actual object from the database belonging to the same type. When the exemplar contains different instances belonging to the same type e.g., two cups, we associate

Figure 3.1: A simple 2D annotation tool (top left) converts photos of real arrangement (left, in each pair) into learning exemplars (right, in each pair).

them with maximally dissimilar instances of the same type in the database, to maximize interest. Similarity is measured using Chamfer distance [20].

While theoretically this step can be automated using computer vision techniques, we found manual annotation to be simple and fast enough for our purposes. Note that an image has to be annotated only once, and can then be used to generate any number of output arrangements with no manual intervention.

3.2 Arrangement Synthesis

3.2.1 Style Measures

Although it is easy for humans to recognize style in many fields, defining it precisely is a difficult problem. For the purpose of populating artifact arrangements, we define style using a combination of object-level and global indicators.

Object-level indicators address placement of individual objects by answering three key questions: 'What?', 'How?', and 'Where?' 'What' determines which types of objects to include in a synthesized arrangement and their distribution; 'How' determines the relations between these objects such as adjacencies and distribution of stacks, and 'Where' determines the relative placement of objects on the support surfaces. Given an exemplar arrangement $\mathbf{S}^*$ and new synthesized arrangement $\mathcal{S}$, we use three scale-invariant measures $f_i(\mathcal{S}, \mathbf{S}^*)$, discussed in Section 4.1, to capture these properties. To simplify the evaluation we assume that pairwise-immediate adjacency relationships between objects are bijective, i.e. that we can have at most one object to the left/right, front/behind, or above/below another one. While this assumption prevents us from handling cases such as two cups placed on one plate, it greatly simplifies the computations and does not significantly affect the realism and visual appeal of the generated results.

In addition to the object-level criteria, we use global scores $g_j(\mathcal{S})$ that evaluate an arrangement as a whole, in terms of design-level characteristics such as organization and symmetry (Section 4.2). The more similar two arrangements look overall, the closer their scores will be. Both the object-level and global functions are normalized to be in the range of $[0, 1]$, with lower values of f_i reflecting more similar arrangements and lower values of g_j reflecting arrangements better conforming with high level design criteria.

3.2.2 Valid Space

We aim for target arrangements which are similar in nature to the exemplar but not too similar. Thus, instead of minimizing some combination of the measures f_i and g_i above, we define a set of inequality constraints corresponding to each measure that together delimit a *valid space* of possible solutions. We use a single scalar $C \in [0, 1]$ to govern the size of the valid space; the object-level differences between the exemplar and target arrangement are required to satisfy $f_i(\mathcal{S}, \mathbf{S}^*) \leq C$. In all the examples we consider in the paper we set $C = 0.25$. In theory, the same formulation can be used to enforce similarity in terms of global characteristics $\|g_j(\mathcal{S}) - g_j(\mathbf{S}^*)\| \leq C$. In practice, we found that a simpler parameter-free condition, requiring the target arrangement to conform with the design criteria at least as well as the exemplar, performs equally well: $g_j(\mathcal{S}) \leq g_j(\mathbf{S}^*)$. These constraints could be considered as defining the barriers of the valid space. We restate these inequalities using barrier functions and use the sum of differences of all

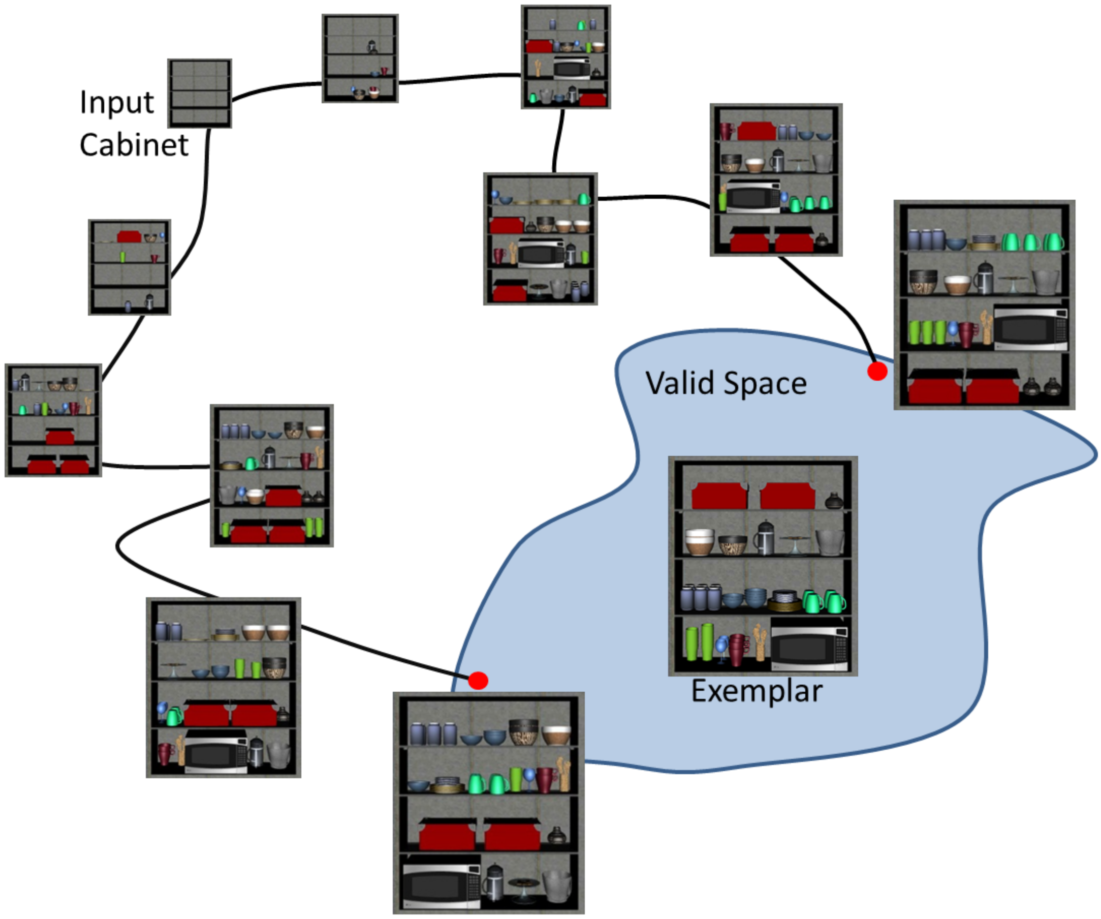


FIGURE 3.2: Illustration of our algorithm: starting from an empty cabinet, a stochastic optimization procedure reaches different valid result inside the valid space (light blue), distinct from the exemplar.

barrier functions from their respective barrier as the global evaluation function:

$$\begin{aligned} \mathrm{eval}(\mathcal{S}, \mathbf{S}^*) = & \sum_i \max\{(f_i(\mathcal{S}, \mathbf{S}^*) - C), 0\} + \\ & \sum_j \max\{(g_j(\mathcal{S}) - g_j(\mathbf{S}^*)), 0\} \end{aligned} \tag{3.1}$$

Using max prevents us from continuing to optimize a function if its barrier is satisfied, and terminates optimization when all barriers are satisfied.

3.2.3 Optimization

Given an exemplar arrangement, processed as described in (Section 3.1), and a target set of surfaces to populate (that hereafter will be called a "cabinet"), we use a randomized synthesis framework to generate output arrangements that minimize Equation 3.1. The synthesis begins with empty shelves and uses the ILS stochastic optimization framework [21, 22]. We found that ILS outperformed more commonly used approaches such as simulated annealing, both in terms of convergence speed and result quality.

ILS starts by finding a local minimum of the given evaluation function using iterative first improvement, a local search technique that loosely resembles gradient descent. It then iterates over a two-phase procedure: first, a perturbation is applied to escape from the current local minimum, and next, iterative first improvement is carried out until another local minimum is reached. If this new local minimum is better than the previous one, the search continues from the new one. Otherwise, it starts the next perturbation phase from the previous. The perturbation stage consists of 5 randomly chosen local modifications selected out of the following set.

- **Add** an object of a type present in the exemplar to a previously empty position in the current arrangement, i.e. a position to the left, top, or behind an input object.
- **Remove** an object from the current arrangement.
- **Replace** an object in the current arrangement with another object of a type present in the exemplar.
- **Move** an object from any position to a new empty position (including stacking it on top of another object).
- **Swap** the positions of two individual objects.
- **Swap** the positions of two columns or stacks of objects (a column is a set of objects defined by the in-front-of relationship).

The modifications are applied only if they are feasible in terms of spacing, object sizes, etc. For instance, whenever an object is placed at a new position we first check if enough space is available in that location to accommodate it. If not, the operation is aborted.

The stochastic search continues until either all the inequalities are satisfied, i.e. the optimized functional (Equation 3.1) is zero, or a maximal number of iterations with no improvement is reached (20 in our setup). The randomized search combined with the validity termination criterion ensure that any repeated execution of the algorithm leads to different results (Figure 3.2), enabling the creation of diverse style-conforming arrangements from a single exemplar.

3.2.4 Fine Tuning Positions

To make the optimization process faster, the adjacency relations considered only follow the three major directions. As a result this procedure tends to generate structured, grid-spaced arrangements. To achieve the desired natural looking arrangements, after the optimization terminates we perturb the positions of individual items, redistributing them along the width and depth directions. We also perturb the orientation of objects with near-rotational symmetry, such as cups. While these steps could be performed as local perturbations within the optimization framework, doing so would in our experience significantly slow the algorithm.

Chapter 4

Style Measures

4.1 Object Level Style Properties

We use three main questions to help define the style of an arrangement: 'What objects are used in the arrangement?', 'How are they placed relative to each other?', and 'Where are they placed in the cabinet?'. To define scale-invariant measures that answer these questions we use the same general functional template, normalizing it to the range $[0, 1]$. The key observation we make is that given two corresponding property values measured on the two arrangements, viewers care about the relative rather than absolute difference between them. For instance, if one arrangement has one microwave and the other has two, it is perceived as a big difference (a factor of two increase), while if one has seven cups and the other has eight, the difference is perceived as minor. Thus given any two values, n_0 measured on $\mathbf{S}^*$ and n measured on $\mathcal{S}$, we first measure the relative difference between them as:

$$d(n, n_0) = \max\left(\frac{n + \epsilon}{n_0 + \epsilon}, \frac{n_0 + \epsilon}{n + \epsilon}\right) - 1. \tag{4.1}$$

Both n and n_0 are always non-negative and a small ϵ is added to both values to avoid division by zero. Note that $0 \leq d(n, n_0) < \infty$. Next, we map this value to a smooth error measure between zero and one using a Gaussian function:

$$\mathrm{err}(n, n_0) = 1 - e^{-\frac{d(n, n_0)^2}{2\sigma^2}}, \tag{4.2}$$

where σ reflects the local sensitivity of the error measure. The larger σ is, the smaller the error for differing values of n and n_0. In many cases we need to compare multiple values of n and n_0, e.g. measured per shelf or per object type. While in some cases averaging the values across all instances can provide a fair assessment, we often want to penalize outlier values, as a single outlier can make the entire arrangement look unnatural. Consider for instance a kitchen shelf where a microwave is placed on top of a cup - no matter what the rest of the arrangement looks like, the result will look artificial. Thus, when summing up the values we weigh them by a function that gives higher weight to poor local measures:

$$\begin{aligned} f_i(\mathcal{S}, \mathbf{S}^*) &= \sum_{n,n_0 \in \Omega_i} w(n, n_0) \cdot \mathrm{err}(n, n_0) \\ w(n, n_0) &= \\ &= (1 + \mathrm{err}(n, n_0))^p / \sum_{n', n'_0 \in \Omega_i} (1 + \mathrm{err}(n', n'_0,))^p \end{aligned} \tag{4.3}$$

where Ω_i is the domain of the specific measure (e.g. all shelves, or all objects), defined differently for each property function f_i. The parameter p controls the overall sensitivity to outliers, necessary since basic averaging over a large set of values can dilute even high local errors.

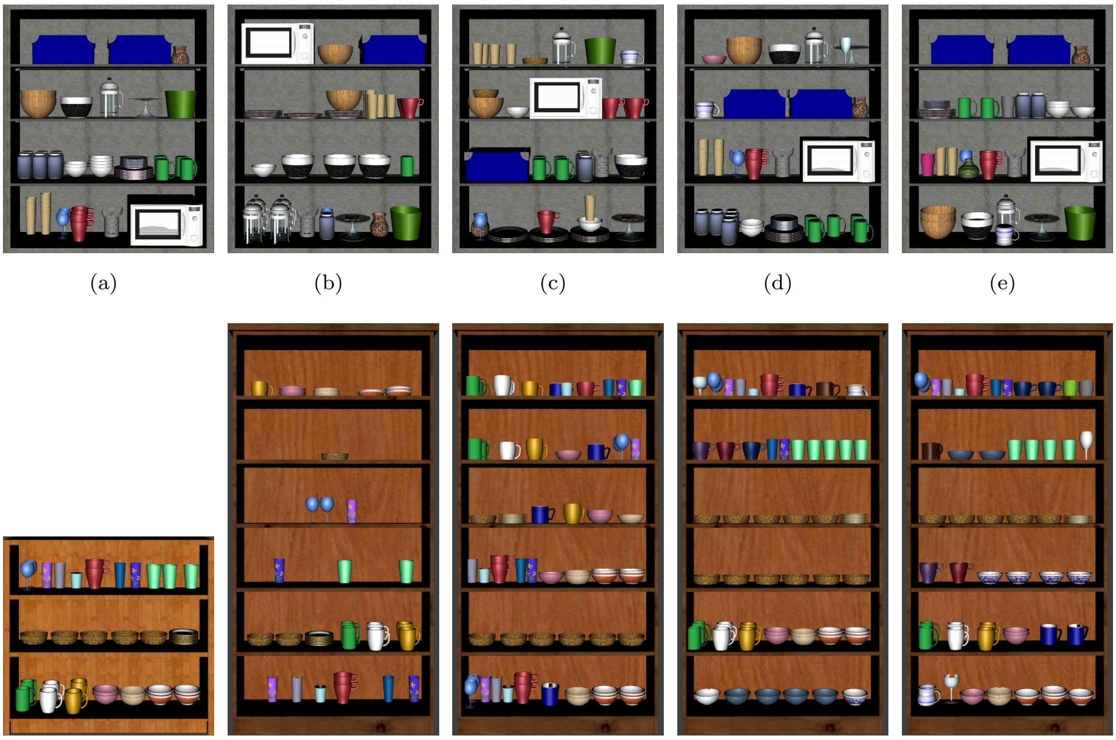
4.1.1 Distribution of Objects (‘What?’)

To measure how well the content of an arrangement $\mathcal{S}$ reflects that of the exemplar $\mathbf{S}^*$, we consider all objects, or labels, present in the exemplar. For each label l, n_0 is the number of objects with the label l divided by the total number of objects in $\mathbf{S}^*$, and n is the number of objects with the corresponding label in $\mathcal{S}$ divided by the total number of objects in $\mathcal{S}$. The sum in Equation 4.3 runs over all labels in the two arrangements. Both σ and p are set to one in the normalized overall measure.

4.1.2 Objects Adjacencies ('How?')

The immediate adjacencies of objects in real-life arrangements reflect both functionality and design. For example, placing similar objects together provides a sense of order, while placing one object on top or behind another reflects both physical constraints and access frequency. We treat each adjacency direction separately, as they are affected by different functional and aesthetic considerations. For each pair of objects we count the number of times the first object shows up *to the right, behind,* or *above* the second one and divide this number by the total number of adjacencies in the corresponding direction. We use the obtained ratios for every pair in the exemplar and input arrangements as the n and n_0 values in the measurement template, summing those separately for each of the three directions. We use $\sigma = p = 1$ for width and depth directions. For stacks we distinguish between the case where an adjacency is less frequent in the output than in the exemplar ($n \leq n_0$) where we use the standard setting of $\sigma = p = 1$ and the complementary case $n > n_0$ where we use $\sigma = 2$ and $p = 10$ to penalize stacking of objects that are rarely or not at all stacked in the exemplar.

In addition to immediate adjacencies our 'how' term compares stack heights between the exemplar and the output arrangement, indicating a preference to stack objects if they are stacked in the exemplar, but not to "over-stack" them. For each object in the exemplar, we use the mean of stack heights of stacks that include this object as n_0 and compare it to n, which is defined as the height of each stack that contains an object with the same label in $\mathcal{S}$. We use the standard setting of $\sigma = p = 1$ to sum up the height differences. We use the mean to avoid bias created by rare stacking choices in the exemplar. E.g. consider a cabinet with rows of unstacked cups and one cup stack. Using distance to mean will result in largely unstacked outputs, while distance to closest existing stack may lead to a multitude of stacked cups.



(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

FIGURE 4.1: The impact of different style function term: both object level (top), and global (bottom). (a) Exemplar; (b) optimizing using only global properties; (c) adding distribution term; (d) adding adjacency; (e) adding positioning. (f) Exemplar; (g) optimizing to a larger cabinet using only object level properties; (h) adding density term; (i) adding grouping and symmetry properties; (j) adding variability.

4.1.3 Objects Placement ('Where?')

Depending on the type of arrangement, objects can have stronger or weaker positional constraints. Intuitively, the position of each object relates to its accessibility and functionality. For example, more commonly accessed items are placed at the front of shelves, heavier objects placed lower and so on. To measure difference in location while accounting for changes in cabinet sizes, we use relative positioning of objects within a cabinet instead of absolute positions. For an object o whose coordinates are (x', y', z') in a given cabinet with bounding box dimensions (b_x, b_y, b_z), we define

$$\text{Pos}(o) = (x'/b_x, y'/b_y, z'/b_z). \tag{4.4}$$

To compare the relative positions of objects in $\mathbf{S}^*$ and in $\mathcal{S}$ we first find for each object

o in the output arrangement the closest object with the same label in the exemplar. Then, we simply use the Euclidean distance between their positioning as the error:

$$\mathrm{err}(o, \hat{o}) = ||\mathrm{Pos}(o) - \mathrm{Pos}(\hat{o})||.$$

We aggregate this error for all objects in the output using $p = 0$, as this metric is not sensitive to outliers. The position of an object is measured as the position of its center of mass.

4.2 Global Properties

The overall style of an arrangement $\mathcal{S}$ can be measured using a number of global characteristics. These characteristics can also be used to compare two arrangements. We use four such characteristics $g_j(\mathcal{S})$ in our formulation. We include a measure for the density of the arrangement, critical for generating arrangements whose content reflects the cabinet size. We measure how grouped (or not) similar items are, and how symmetrically they are arranged. Both of these measures affect the sense of *order* in an arrangement. Lastly, we include a measure for variability of the arrangement, as it affects the sense of *richness* in an arrangement. Figure 4.1 illustrates the effect of adding each term sequentially into the global objective function.

4.2.1 Density

We use two functions to capture the ratio of filled to empty space in a cabinet. The first function examines the overall percent of filled area on shelves, and the second examines the density of the front-row of objects. Usually, the front row is the one that is most visible when viewing cabinets; therefore it is the most important. Let area(o) be the area of the bounding box of an object $o \in \mathcal{O}$. We define:

$$g_{\mathrm{density}}(\mathcal{S}) = 1 - \frac{\sum_{o \in \mathcal{O}} \mathrm{area}(o)}{\mathrm{area}(\mathcal{S})} \tag{4.5}$$

(a) GROUND-TRUTH (b) RAND (c) FULL (d) RAND-OPT (e) Believability survey results

FIGURE 4.2: Example results of the different methods used in our believability survey. The percent each method was marked as "realistic" by viewers is shown in (e). Our optimization method creates believable results that cannot be proven to be significantly different than ground truth, while both RAND-OPT and RAND are (see details in text).

Let $\text{width}(s)$ be the width of a shelf s, and $\text{width}(o)$ width of an object o respectively, and $\text{front}(s)$ the set of objects at the front of the shelf s. We define:

$$g_{\text{width}}(\mathcal{S}) = 1 - \frac{1}{|s|}\sum_{s} \frac{\sum_{o \in \text{front}(s)} \text{width}(o)}{\text{width}(s)} \tag{4.6}$$

where $|s|$ is the number of shelves.

4.2.2 Grouping

This function captures the degree to which identical objects are placed close to each other. Intuitively it measures how many disjoint clusters of identical objects show up in the arrangement, where the ideal number would be a single cluster. To determine $g_{\text{group}}(\mathcal{S})$, we cluster the identical objects (objects that have the same label) placed next to each other in $\mathcal{S}$ using a basic "region growing" algorithm. Given a label l, let $C(l)$ be the number of clusters of label l that appear in the cabinet and $\mathcal{L}_{\text{S}}$ the set of all labels. We define:

$$g_{\text{group}}(\mathcal{S}) = 1 - \frac{|\mathcal{L}_{\text{S}}|}{|\sum_{l \in \mathcal{L}_{\text{S}}} C(l)|} \tag{4.7}$$

To capture the style of $\mathbf{S}^*$ we aim for $g_{\text{group}}(\mathcal{S})$ to be the same as $g_{\text{group}}(\mathbf{S}^*)$.

4.2.3 Symmetry

The symmetry function is formally defined using the notion of mirror objects. Given a shelf s of width w that is part of a given arrangement $\mathcal{S}$, and an object o on s whose position's x-coordinate is x_o, we define its mirror object o' on the same shelf as the object whose position's x-coordinate is closest to $(x_o)' = w - x_o$, the mirror position of o. To evaluate symmetry we measure two things: how similar these two objects are, and how far the mirror object, o', is positioned from the mirror position (the perfect symmetry location). Hence, for each pair of an object o and its mirror object o' we have:

$$\mathrm{d}(o, o') = \frac{1}{3} \cdot ((2 - \mathrm{sim}(o, o')) \cdot (1 + \frac{\|(x_o)' - x_{o'}\|}{w}) - 1)$$

where the similarity $\mathrm{sim}(o, o')$ is measured using Chamfer distance. By offsetting both terms, we ensure that they do not cancel each other, and the final multiplication by $\frac{1}{3}$ scales the value of the function to the range $[0, 1]$. We define the symmetry measure of an arrangement $\mathcal{S}$ as

$$g_{\text{symmetry}}(\mathcal{S}) = \frac{1}{|s|} \sum_{s} \frac{\sum_{o \in \mathcal{O}} \mathrm{d}(o, o')}{|\mathcal{O}|}. \tag{4.8}$$

4.2.4 Variability

To estimate the diversity of an arrangement we measure two factors: how many distinct object labels are used in the arrangements L compared to the overall number of objects O, and how evenly the objects are distributed among them. To measure the latter, we look at the number of objects assigned the least used label $o_{\min}$ and the number of objects with the most used one $o_{\max}$. We define:

$$g_{\text{variability}}(\mathcal{S}) = (1 - \frac{L}{O}) \cdot (1 - \frac{o_{\min}}{o_{\max}}). \tag{4.9}$$

Chapter 5

Validation

We validate our arrangement approach using both a visual evaluation, discussed in Section 6, and a more quantitative one performed via a user study. The goal of the study was to validate the key properties of our arrangement technique, proving that the arrangement results produced by our method are *believable*, i.e. similar to the arrangements of objects one would expect to find in a real house, and confirming that the method can capture the *style* of a given input exemplar arrangement. To validate scalability we test both properties across differently sized and shaped cabinets.

Our study was conducted via Amazon Mechanical Turk and used duplications of questions in random order to filter out inconsistent answers (different responses to the same question) and inconsistent responders (those who gave too many different answers to duplicate questions).

5.1 Believability

User responses as to whether an arrangement is believable depend not only on arrangement content and layout but also on extraneous parameters, such as rendering quality and the degree of realism of the 3D models used. To maximally control for other factors, we showed users a random mixture of results generated using four different methods, but using the same objects and rendering tools (see examples in Figure 4.2). We then asked

Our method with average precision of 84%				
	O	**S**	**G**	**U**
Original	79.67%	4.95%	5.49%	9.89%
Symmetric	2.39%	91.39%	3.83%	2.39%
Grouped	5.36%	7.14%	84.52%	2.98%
Unorganized	8.78%	4.39%	4.39%	82.44%
Mixed	14.97%	8.38%	4.79%	71.86%

Using global properties - average precision of 39%				
	O	**S**	**G**	**U**
Original	43.56%	14.85%	31.19%	10.40%
Symmetric	7.59%	71.43%	13.84%	7.14%
Grouped	10.18%	18.14%	34.07%	37.61%
Unorganized	10.48%	21.90%	59.05%	8.57%

FIGURE 5.1: The confusion matrices of people's classification of styles using our method (top) and only global properties (bottom). The rows show the true style of the synthesized results (**M**ixed are synthesized results that use all style input exemplars), and the columns denote the style classified by users. When using only global properties, the average precision drops from 84% to 39%. Similarly, trying to learn from multiple styles (bottom row in top matrix) produces an unorganized style.

FIGURE 5.2: Examples from our style study. The top row shows style exemplars (from left): **O** - Original exemplar from an image, **S** - symmetric style, **G** - grouped style, and **U** - unorganized style. The bottom row shows typical results generated by our method based on these exemplars.

them to specifically evaluate the realism of each arrangement ignoring other factors. In addition to results generated by our method (FULL), we included ground truth results (GROUND-TRUTH) generated by converting real images into 3D arrangements using the annotation process in Section 3.1, and randomized arrangements (RAND) created by placing objects randomly in the cabinet, selecting uniformly at random from the objects contained in the exemplar. We also included a simplified version of our method (RAND-OPT) which starts with generating a randomized arrangement and then optimizes only the global terms from Section 4.2. The RAND-OPT approach can be seen as a adaptation of rule-based layout techniques such as [3] to artifact arrangement, one where the rule parameters are learned from the exemplar. The goal here was to test if such rule based approaches are sufficient to achieve believable results in our setup.

First Hypothesis: For all three synthesis methods, the null hypothesis states that there is no significant difference between the frequency with which people will evaluate the arrangement results of these methods as ‘real’, and the frequency with which they will evaluate ground truth arrangements as ‘real’.

Study 1: We created arrangements based on the four methods, using various exemplar images and different cabinet sizes, and included an equal number of results from each method, in each questionnaire. We gathered 183 consistent participants in this study using a 70% consistency threshold. We used the Chi-square test with one degree of freedom and for both RAND-OPT and RAND we could reject the null hypothesis with a level of significance < 0.05, while we could *not* reject the null hypothesis for our FULL method. The study results, summarized in Figure 4.2, confirm that our optimization produces believable results that are almost indistinguishable from ground truth data and that significantly outperform the results of the randomized method.

5.2 Style

To confirm that our approach captures the input arrangement style, we evaluated people’s ability to recognize style by matching the outputs of our algorithm to the exemplars they were created from. We used two sets of exemplars with two different contexts, a kitchen and a living room. For each set we used four cabinets (different from the exemplar cabinets) to synthesize new results. To generate distinct styles, we picked a fairly non-nondescript real image and generated three variations on it - one highly symmetric, one highly grouped, and one highly unorganized (Figure 5.2). We generated results using both our full method (FULL) and the rule based method (RAND-OPT) lacking the object-level optimization terms. To avoid recognition bias, in the synthesized results we used objects of the same type but different from the objects found in the exemplar.

Second Hypothesis: We hypothesize that our full method - based on both object-level and global properties of an arrangement captures well the style of a given arrangement and performs better than optimizing global properties alone - i.e. approximating rule-based assembly approaches.

Study 2: We devised a classification test where in each question the four input exemplars (the choices) were shown to the participant along with one synthesized result (the query arrangement). The participant was asked to decide which of the four choices looked more similar in terms of its style to the given query arrangement. Because of multiple choices in this study, we used a 60% consistency threshold, arriving at 65 consistent participants for FULL and 76 for RAND-OPT. Figure 5.1 summarizes the confusion matrix of the various styles. Our method provides 84% precision overall, a fairly impressive number, especially when compared against the 39% of the baseline RAND-OPT approach, mimicking existing rule-based layout methods.

5.3 Variability

A major goal of our valid space approach is to enable creation of diverse arrangements for the same, or similar, cabinets using a single exemplar. To quantify the variability of our outputs we generated three series of results each from a different exemplar, each populating the same cabinet as the exemplar. We then measured the pairwise differences between all pairs of results within each series using the objects placement metric from Section 4.1 (Eq. 4.4). Using the default parameter $C = 0.25$ to control the valid space size, the average pairwise distance between results was 0.19 (see Figure 5.4). To

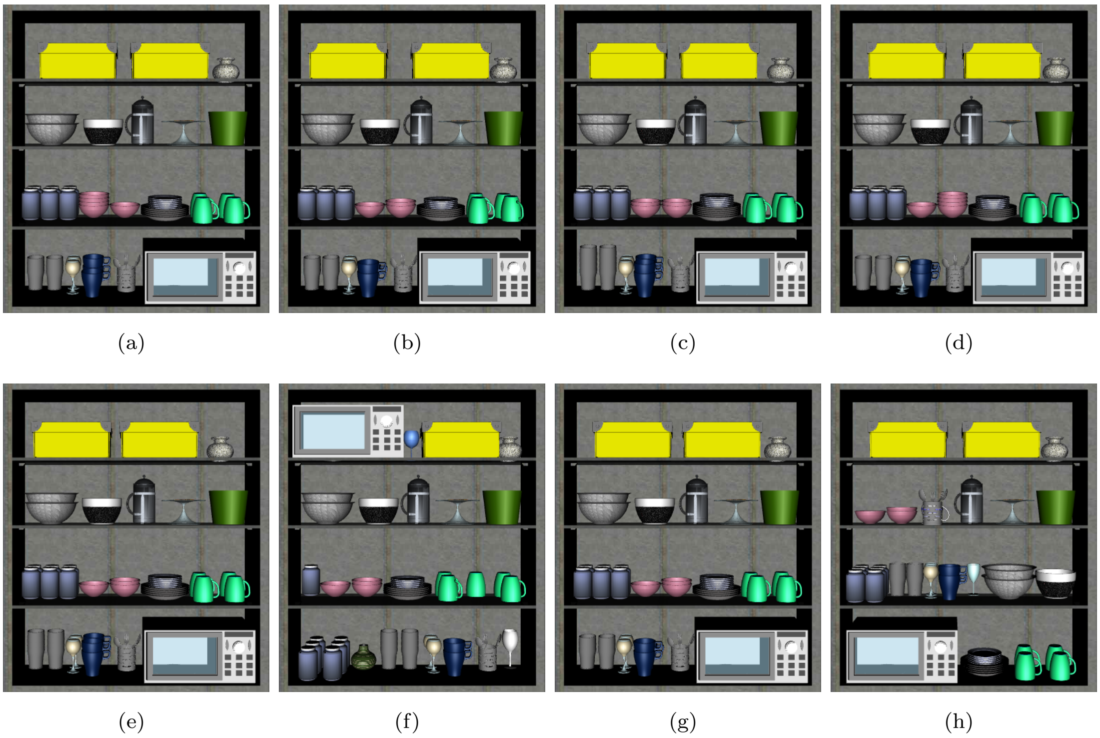


FIGURE 5.3: Results of eight different executions of running the optimization with no valid space (terminating when reaching a stable minimum). Note how results are very close and very similar to the input exemplar (compare to Figure 5.4).

mimic optimization techniques that converge to a local (global) minima of a distance to exemplar function, employed e.g. for furniture arrangement [4–6], we also ran our method until full convergence (using $C = 0$). In this case, even though the optimization process remained randomized creating some variation, the difference measured within a series of results plummeted, with the average distance going down to 0.04, signifying reduced variability (see Figure 5.3 and compare to Figure 5.4).

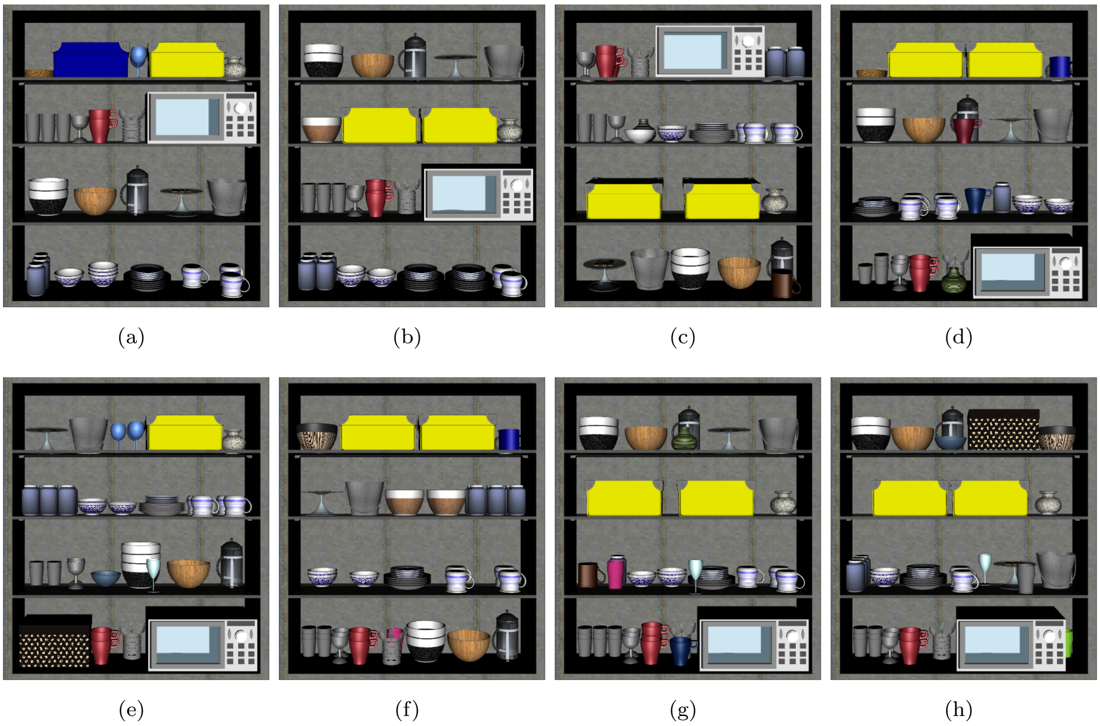


FIGURE 5.4: Results of eight different executions of running the optimization terminating once the solution is inside the valid space (using a threshold of 0.25). Note the variability of results while still preserving believability and similar style (compare to Figure 5.3).

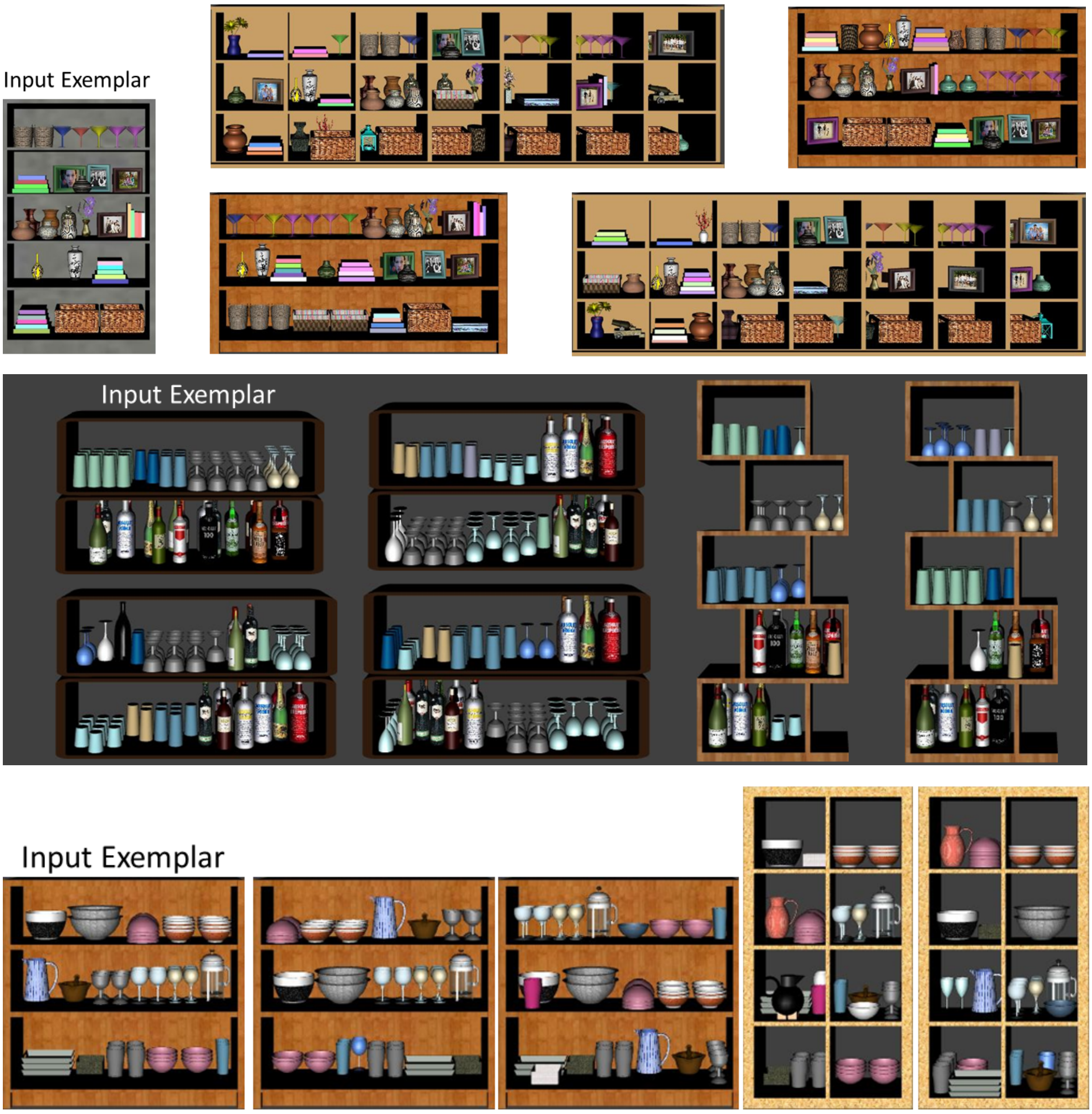


FIGURE 5.5: Examples of various arrangements capturing exemplar style (left) for differently sized cabinets. To emphasize arrangement level rather than object variability we use exactly the same objects for all cabinets.

Chapter 6

Results

Figures 1.1 through 6.1 show the results generated by our method on a variety of inputs. Optimization times for all results were under 5 seconds per cabinet. Figures 1.2 and 5.2 showcase our ability to generate different style arrangements for the same cabinet. The exemplars used for Figure 1.2 are shown in Figures 5.5 and 4.1. Figure 5.5 highlights our method's ability to generate various scalable arrangements for different contexts: a living room cabinet, a bar, and a kitchen cabinet. The larger output cabinets demonstrate our ability to retain believability and style while doubling arrangement size. To emphasize arrangement level rather than object type variability, we use exactly the same mapping of objects to labels for each sequence of exemplar and outputs in this figure. In practice, users can introduce more diversity by varying the label to object choices (e.g. selecting a different cup to correspond to the CUP1 label in the exemplar), as demonstrated in Figures 1.1 and 6.1. Figure 6.1 further highlights the range of styles and contexts we capture. In addition to the visual inspection, we measured output variability as discussed in Section 5.3. The average value of outputs produced by our method (with $C = 0.25$) in these experiments remained around 0.2 across the different inputs, providing quantitative validation of our approach.

FIGURE 6.1: Arrangement medley: shown are the input exemplars and sample output arrangements for different cabinets and sizes (randomizing the objects used for each input label)

6.1 Limitations

Learning from one exemplar is still a challenging problem and our main limitation is the inability to learn "general" rules that many times govern an arrangement such as the settings of a dinner table. Such a new setup can be based on different functional considerations than those we defined, ones that cannot be learned from the exemplar alone. However, we believe that with suitable definition of local and global measures our approach can be extended to other arrangement problems as well. Another current limitation of our approach is the directional adjacency relations we assumed when assembling our arrangements. This prevents us, for instance, from positioning two glasses on or behind one plate. The fine-tuning step relaxes this constraint, used when updating arrangements during the optimization. Enhancing the set of possible perturbations can resolve this limitation, but would make the updates more complex.

Chapter 7

Conclusions

We presented a method for generating believable arrangements of artifacts laid out on different support surfaces. Using a single style exemplar our method creates a variety of style-preserving arrangements, scaling to different output cabinet sizes. We validated the method, testing it on a variety of inputs and confirming style-preservation and believability via a user study. The key concept behind our method is the use of valid-space to find solutions which are both style-preserving and diverse. This approach enables us to generate numerous arrangements close enough, but not too close, to the input exemplar. A similar method could potentially be used for other setups where both variability and style are important - e.g. for synthesizing new shapes or even character motions.

Bibliography

[1] P. Merrell, E. Schkufza, and V. Koltun, "Computer-generated residential building layouts," *ACM Trans. Graph.*, vol. 29, pp. 181:1–181:12, December 2010. [Online].

[2] T. Kelly and P. Wonka, "Interactive architectural modeling with procedural extrusions," *ACM Trans. Graph.*, vol. 30, pp. 14:1–14:15, April 2011.

[3] P. Merrell, E. Schkufza, Z. Li, M. Agrawala, and V. Koltun, "Interactive furniture layout using interior design guidelines," *ACM Trans. Graph.*, vol. 30, pp. 87:1–87:10, August 2011. [Online].

[4] L.-F. Yu, S.-K. Yeung, C.-K. Tang, D. Terzopoulos, T. F. Chan, and S. J. Osher, "Make it home: automatic optimization of furniture arrangement," *ACM Trans. Graph.*, vol. 30, pp. 86:1–86:12, August 2011. [Online].

[5] Y.-T. Yeh, L. Yang, M. Watson, N. D. Goodman, and P. Hanrahan, "Synthesizing open worlds with constraints using locally annealed reversible jump mcmc," *ACM Trans. Graph.*, vol. 31, no. 4, pp. 56:1–56:11, 2012.

[6] M. Fisher, D. Ritchie, M. Savva, T. Funkhouser, and P. Hanrahan, "Example-based synthesis of 3d object arrangements," in *ACM SIGGRAPH Asia 2012 papers*, ser. SIGGRAPH Asia '12, 2012.

[7] B. Harris, "How to arrange shelves with style, http://www.houzz.com/ideabooks/9874/list/how-to-arrange-shelves-with-style."

[8] L. Hennen, "Decorating tips for shelves and bookcases, http://www.hgtv.com/decorating-basics/decorating-tips-for-shelves-and-bookcases/pictures/index.html."

[9] M. Fisher, M. Savva, and P. Hanrahan, "Characterizing structural relationships in scenes using graph kernels," *ACM Trans. Graph.*, vol. 30, pp. 34:1–34:12, August 2011. [Online].

[10] G. Chen, G. Esch, P. Wonka, P. Mueller, and E. Zhang, "Interactive procedural street modeling," *ACM Trans. Graph.*, vol. 27, no. 3, pp. Article 103: 1–10, 2008.

[11] M. Lipp, D. Scherzer, P. Wonka, and M. Wimmer, "Interactive modeling of city layouts using layers of procedural content," *Computer Graphics Forum (Proc. EG 2011)*, vol. 30, no. 2, pp. 345–354, Apr. 2011. [Online].

[12] J. Lin, D. Cohen-Or, H. R. Zhang, C. Liang, A. Sharf, O. Deussen, and B. Chen, "Structure-preserving retargeting of irregular 3d architecture," *ACM Transactions on Graphics (Proc. of SIGGRAPH Asia 2011)*, vol. 30, no. 6, dec 2011.

[13] C. Ma, L.-Y. Wei, and X. Tong, "Discrete element textures," *ACM Trans. Graph.*, vol. 30, pp. 62:1–62:10, August 2011. [Online].

[14] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. Efros, "What makes paris look like paris?" *ACM Trans. Graph.*, 2011.

[15] K. Xu, H. Li, H. Zhang, D. Cohen-Or, Y. Xiong, and Z. Cheng, "Style-content separation by anisotropic part scales," *ACM Transactions on Graphics, (Proc. of SIGGRAPH Asia 2010)*, vol. 29, no. 5, 2010.

[16] M. Agrawala, D. Phan, J. Heiser, J. Haymaker, J. Klingner, P. Hanrahan, and B. Tversky, "Designing effective step-by-step assembly instructions," *ACM Transactions on Graphics (TOG)*, vol. 22, no. 3, pp. 828–837, 2003.

[17] J. Kopf, M. Agrawala, D. Bargeron, D. Salesin, and M. F. Cohen, "Automatic generation of destination maps," *ACM Transactions on Graphics (Proc. of SIGGRAPH Asia 2010)*, vol. 29, no. 5, pp. 158:1–158:12, 2010.

[18] M. Agrawala, W. Li, and F. Berthouzoz, "Design principles for visual communication," *Communications of the ACM*, vol. 54, no. 4, pp. 60–69, April 2011.

[19] M. L. Van, *Shelf Expression: 70 Projects & Ideas for Creative Storage & Display.* Lark Books, 2004.

[20] D. Coquin and P. Bolon, "Discrete distance operator on rectangular grids," *Pattern Recognition Letters*, vol. 16, no. 9, pp. 911–923, 1995.

[21] H. R. Lourenço, O. Martin, and T. Stützle, "Iterated local search," in *Handbook of Metaheuristics*, F. Glover and G. Kochenberger, Eds. Kluwer Academic Publishers, Norwell, MA, USA, 2002, pp. 321–353.

[22] H. H. Hoos and T. Stützle, *Stochastic Local Search—Foundations and Applications.* Morgan Kaufmann Publishers, 2004.

[23] E. Kalogerakis, S. Chaudhuri, D. Koller, and V. Koltun, "A Probabilistic Model for Component-Based Shape Synthesis," *ACM Trans. Graph.*, vol. 31, no. 4, pp. 384–394, 2012. [Online].

תקציר

הבתים וחללי העבודה שלנו מלאים בחפצים רבים הפזורים על גבי משטחים כמו מדפים ושידות. התוכן וצורת הפריסה של סידור החפצים משקף את ההקשר בו הם נמצאים (למשל מטבח, סלון, עיצוב מסודר או לא מסודר).
סידור ידני בסצנות וירטואליות מבזבז זמן רב, בייחוד כאשר יש צורך לייצר סידורים שונים ומגוונים עבור מספר רב של משטחים וחללים.
אנו מציגים שיטה מונחית נתונים המיועדת לסידור חפצים ופיזורם באופן אוטומטי על משטחים ריקים באופן משכנע ובסגנון נתון.
אמצעי הקלט בשיטה שלנו הוא תמונה או מודל תלת מימדי של סידור לדוגמה, המשקף את התוכן והסגנון המתבקש. השיטה שלנו עושה שימוש בדוגמה הזו על מנת לייצר סידורים מגוונים המשקפים את הסגנון שלה עבור סידור של ריהוט ומימדי פריסה.
על מנת להשיג בו זמנית יכולת הרחבה, גיוון ושימור סגנון, אנו מגדירים מרחב פתרונות אפשריים של סידורים המשקפים את סגנון אמצעי הקלט.
אנו משיגים פתרונות בתוך המרחב הזה באמצעות שימוש בפונקציות מחסום ואופטימיזציה סטוכסטית.

הבינתחומי הרצליה
בית ספר אפי ארזי למדעי המחשב

המרכז הבינתחומי הרצליה
בית ספר אפי ארזי למדעי המחשב

מילוי מדפים: יצירת סידור מגוון של חפצים תוך שמירה על סגנון

עבודת תזה המוגשת כמילוי חלק מהדרישות לקראת תואר מוסמך במסלול מחקרי במדעי המחשב

על-ידי לוקס מחרוביץ'

העבודה בוצעה בהנחיית פרופ' אריאל שמיר

דצמבר, 2014