



南京大學

研究生畢業論文
(申請碩士學位)

論文題目 数据驱动的室内场景着色方法研究

作者姓名 马晗

学科、专业方向 计算机科学与技术

指导教师 郭延文 教授

研究方向 计算机视觉

2017 年 5 月

学 号 : MG1433045
论文答辩日期 : 2017 年 5 月 25 日
指 导 教 师 : (签字)

Research on Data-driven Approach for Furniture and Indoor Scene Colorization

by

Han Ma

Directed by

Professor Yanwen Guo

Computer Science Department

Nanjing University

May 2017

*Submitted in partial fulfilment of the requirements
for the degree of Master in Computer Science and Technology*

南京大学研究生毕业论文中文摘要首页用纸

毕业论文题目: 数据驱动的室内场景着色方法研究
计算机科学与技术 专业 2014 级硕士生姓名: 马晗
指导教师(姓名、职称): 郭延文 教授

摘 要

针对一个由多个家具模型组成的室内场景,目前关于模型的布局摆放等问题的研究已经相对成熟,但关于家具模型以及场景的自动着色还没有很好的进展。而一个场景给人最直观的感觉就来源于颜色及其搭配,因此模型着色对于构建一个美观和谐的三维场景而言是至关重要的。但是对于一般人而言要选择每个物体的颜色是一个繁琐且耗时的任务,而且也难以选择出完美的颜色搭配,使得整个场景的颜色风格看起来协调一致。就算是专业的室内装饰设计师或艺术家可以根据其丰富的经验以及直觉来选择配色,为模型逐个上色也是一件麻烦的事情。如果能够实现一套为三维模型场景自动着色的方法,则不仅可以用来模拟生成仿真的室内场景,为装饰设计提供参考,也可以用于游戏场景构建,现实场景模拟等图形方面的应用。

对于一个场景的着色,除了对每个物体进行着色,还要注意整个场景中物体之间颜色的搭配是否和谐,实际上对于一个物体选定其材质容易,但进一步选定材质对应的颜色却有多种选择。而另一个难题是许多模型并非是结构化的,虽然模型可以划分为一组拓扑独立的组件,但并没有语义信息,而且组件过多且复杂,对模型组件的颜色选取不能仅根据其功能性。所以颜色的选取和模型的分割是两个主要需要解决的问题。

本文提出了一种基于数据驱动的方式为室内场景自动着色的方法,该方法包含以下几个方面:首先我们建立了一个数据库,包括图片-模型数据库和材质数据库两部分。我们从网络中收集了大量由设计师设计或摄影师拍摄出来的室内场景图像,以及一些家具物体的三维模型。为了建立图片与模型之间的对应关系,我们设计了一个用户友好的图片标识软件,可以更方便地完成图片中家具物体及其组件标识的工作,从而将标识细化到每个物体的组件部分,使得模型与图片中的物体在组件级别对应起来,构建了一个包括场景层,物体层和组件层三个层次的层级式图片-模型数据库。我们利用了一种新的材质表面信息获取方法构建了一个高质量的材质库,该方法结合多组拍照技术来提取材质表面的信息,为我们渲染出逼真的室内场景提供了基础。

其次本文实现了对单个家具模型的着色。输入为一个三维家具物体模型和一张相同类型家具的参考图片，根据我们上一步骤构建的层级式图片-模型数据库，我们可以直观地找到图片与模型之间的对应关系，并以图片的标识信息作为参考来对家具模型进行分割。我们提取了一系列网格级和组件级的特征，对每一种分割方式进行学习从而得到图片引导的模型分割分类器，来对输入模型进行自动分割。通过将分割结果与图片中物体的不同颜色部分对应起来，我们在材质数据库中搜索与图片中家具物体的各个部分颜色最相近的材质,就可以容易地为三维模型赋予与图片相似的着色风格。

最后本文实现了对场景的着色。我们提出了两种着色方式：参考示例图片或颜色主题进行着色。如果以示例图片作为参考，则可以按照对单个家具模型着色的方法为场景的中的每个物体逐个进行着色即可。如果以颜色主题作为参考，我们首先根据着色问题构建出一个马尔科夫随机场模型，并利用数据库中的数据为每个家具物体构建颜色概率分布，然后以用户的输入作为约束条件构建了一个能量方程，通过求解该方程得到每个家具模型的颜色主题，最终生成出与用户期望的颜色主题一致的室内场景着色结果。我们的实验结果以及用户调查表明我们的系统能够生成与室内设计师设计的结果相媲美的着色方案。

关键词： 着色; 室内设计; 数据驱动; 网格分割

南京大学研究生毕业论文英文摘要首页用纸

THESIS: Research on Data-driven Approach for Furniture and Indoor Scene
Colorization

SPECIALIZATION: Computer Science and Technology

POSTGRADUATE: Han Ma

MENTOR: Professor Yanwen Guo

Abstract

For an interior scene with multiple furniture models, there have been progressive improvements to help users to create effective furniture layout in recent years, but colorization for furniture objects and further the whole indoor scene has not yet received the considerable attention that it deserves. The fact is that much of what people perceive and feel about the scene is experienced through colors and their combination. Therefore, colorization of models does matter for building a visually pleasing 3d scene. However, this is a tedious and daunting task to choose the color of every object; it is also difficult to combine the colors of objects to make the whole scene harmonious. Even for experienced interior designer or artist, assigning color to models one by one is fussy. So if we can find a way to colorize 3D scene automatically, not only can we use it for interior design, but also for other applications such as game scene and realism.

To colorize a scene, except assigning color to every object, the combination of all the objects in the scene is also important. In fact determining the texture of an object is easy, but there are many colors can be chosen for one texture. Another problem is that furniture models are often unstructured. Although the furniture model is composed of a set of topologically independent components, each of which is not necessarily to be semantically meaningful. Even though semantic segmentation applies to the model, the object is often not colorized completely according to different functional parts. So the main two problems is the selection of color and segmentation of models.

Here we present a data-driven approach that colorizes 3D furniture models and indoor scenes automatically, including following steps: first of all, we build a database,

including a image-model database and a texture database. We collect a large quantity of works by professional interior designers, also some 3D furniture models from the internet. To build correspondences between images and models, we develop a user-friendly prototype, making the task of furniture and components labelling easier. With the labelling data, we build a well-structured image-model database with detailed annotations and image-model correspondences, including scene level, object level and part level. By using multiple photography skills to extract textures' surface information, we also build a texture database with high quality, which is the base for rendering realistic results.

Furthermore, we implement colorization for single furniture model. With the input of a 3D furniture model and a same type furniture image as reference, according to the database, we can find correspondence between images and furniture models directly. By using color and segmentation information of furniture in images and a serials extracted features at mesh and part level, we can train the classifier with multiple segmentation schemes. After matching the different parts in images and models, we can colorize the furniture easily by searching textures with similar colors.

Finally yet importantly, we implement colorizing whole 3D scene. We provide two circumstances: referring image or color theme. For referring image, we can colorize every model using the approach for colorizing single furniture model. For referring color theme, we first build a Markov Random Field model and color probability distribution for objects of every type, then formulate an energy function with user input as constraint. By optimizing the function, we get color theme for every model and finally colorize the 3D indoor scene resembling user desired color theme. Our experiments and a user study show that our system produces perceptually convincing results comparable to those generated by interior designers.

Keywords: Colorization, Interior Design, Data-driven Approach, Mesh Segmentation

目录

目录	v
第一章 概述	1
1.1 室内场景着色问题及其难点	1
1.2 本文贡献及后续内容安排	3
第二章 相关工作	5
2.1 图片与视频着色	5
2.2 网格着色与场景装饰	6
2.3 室内场景建模和家具布局	7
第三章 数据库的建立	8
3.1 图片-模型数据库	8
3.1.1 图片标定工具	9
3.1.2 图片标识工具介绍	10
3.1.3 图片标识流程	12
3.1.4 图片-模型数据库结构	15
3.2 材质数据库	16
3.2.1 硬件准备	16
3.2.2 图像矫正	16
3.2.3 材质获取	17
第四章 基于数据驱动的着色算法详述	22
4.1 图片引导的家具着色	23
4.1.1 图片引导的家具分割	23
4.1.2 家具模型着色	25
4.2 场景着色	25
4.2.1 按照示例图片着色	26
4.2.2 基于数据驱动的着色框架	26

第五章 实验应用以及用户调查	31
5.1 网格分割	31
5.2 家具和场景着色	32
5.2.1 家具着色	34
5.2.2 场景着色	34
5.2.3 实用性	36
5.2.4 用户调查	38
第六章 总结与展望	41
6.1 本文工作总结	41
6.2 未来工作展望	41
参考文献	43
科研成果	48
致谢	49

表格

5.1	本文中用到的场景的统计数据	33
5.2	我们的方法与室内设计师所花费的时间	39

插图

1.1	没有材质的三维模型场景和真实场景	1
1.2	图片与模型之间的对应关系并不明显	2
1.3	两种场景着色方法	3
3.1	LabelMe开源在线图片标识工具	8
3.2	我们编写的图片标识工具	10
3.3	mask图片	12
3.4	图片标识流程	13
3.5	图片-模型数据库结构	14
3.6	按照图片标识对三维模型分割	15
3.7	材质法线信息图片	18
3.8	材质漫反射信息图片	19
3.9	材质高光信息图片	19
3.10	材质置换信息图片	20
3.11	材质透明度信息图片	20
3.12	对一个办公室场景着色所选择的材质	21
4.1	三维模型和场景着色框架流程图	23
4.2	根据场景构建的无向图模型	27
4.3	按照颜色主题着色的一组会议室结果	29
5.1	图片引导的网格分割结果	31
5.2	使用不同特征的网格分割结果比较	32
5.3	椅子模型着色结果	33
5.4	床模型着色结果	34
5.5	参照示例图片的着色结果	35
5.6	用户输入颜色主题作为约束的着色结果	35
5.7	用户通过交互调整着色结果	36

5.8 橱柜上颜色图案的例子	37
5.9 在Google上使用关键词“chair”搜索到的图片	37
5.10 用户调查评分结果统计	38
5.11 用户调查图片	40

第一章 概述

想象当你面对这样一个场景：你搬入了新家，家具的布置已经大致确定了，你已经谋划好了在这放一个沙发，在那放一个茶几，但面对家具店里琳琅满目的家具时，如何挑选和搭配家具的颜色却可能让你头疼。同样室内设计师和游戏开发者也可能在创建了一个新的场景并确定好场景内物体的布局后面临同样的问题。如图1.1所示，对于一个单纯由没有材质的模型组成的三维场景，如何设计选择颜色材质使其拥有如同真实场景一样的效果并不简单。然而在最近的几年内，关于如何帮助用户有效的创建物体摆放布局的研究已经有了不错的进展[1]，[2]，[3]，但对物体以及整个场景的自动着色问题还没有足够的研究。



(a) 三维模型场景

(b) 真实场景

图 1.1: 没有材质的三维模型场景和真实场景

因此我们需要一种方法来向用户推荐合适的着色策略，使得用户不必再纠结于繁琐的颜色选择。本章后续内容将概略论述该问题中包含了哪些难点，并继而提出我们对于此问题的新的见解，最后再介绍本文所提出的着色方法及其贡献。

1.1 室内场景着色问题及其难点

专业的设计师通常有着几年的室内设计经验，他们凭借着这些经验以及直觉来对物体以及场景进行着色，使得场景从美学的角度以及视觉上赏心悦目。然而这个任务通常是枯燥的而且需要大量的用户交互，尤其是对于一些比较复杂的场景。我们的目的就是实现一种算法，首先能够对家具物体进行着色，其

次能够向用户推荐视觉上赏心悦目的室内场景着色方案。这项技术将不仅用于室内设计方面，也将在其他需要全自动化场景建模的图形学方面以及高度现实模拟方面的应用中发挥作用。

需要注意的是尽管在图片渲染方面选取准确的材质是很重要的，但是人类对一个场景的感知很大程度上受到颜色及其组合的影响。另一方面一个特定的家具物体只有有限的材质可选，比如沙发通常是布质的或皮革的，而餐桌和柜子通常是木质的。然而这些家具都能够被涂上任意颜色。因此要从物理角度上决定不同类型物体的材质是容易的[4]，但决定颜色却很难，尤其是当有很多种选择的时候。



图 1.2: 左：一张包含一个现代沙发的客厅图片。右：拥有40个拓扑不连续组件的沙发模型。很明显这些组件与沙发的不同颜色区域之间的对应关系很模糊。自动将沙发模型着色成与图片中的沙发相似的颜色主题并不容易。

除了颜色选取，模型方面的问题也有待解决。网络中有很多室内场景模型，然而存在的一个主要难题是这些家具模型通常是非结构化的。如图1.2所示，家具模型是由一系列拓扑独立的部件组成的，尽管设计师可能已经在构建几何模型的时候对给个组件都进行了命名，但这些命名在语义上也通常是没有意义的。这就为用户在使用三维建模软件，比如3ds Max，来给模型添加不同的颜色和材质时增添了难度，使得这一过程变得十分繁琐并且耗时，特别是对于那些有大量组件的模型。而且即使实现了模型的语义分割，物体各个部分的着色也并不是完全与其功能性相对应的，用户需要准确地挑选出那些具有相同材质的组件部分，并赋予相同的材质。

解决这一问题的关键是自动建立这些组件与有着不同颜色的图片中的物体组件之间的对应关系。这可以归类为一个标定问题。我们首先建立了一个图片-模型数据库，通过该数据库建立图片中的物体与模型之间的对应关系，以图片中标定的结果作为模型分割的指引，然后学习图片引导的网格分割，一旦学习

完毕，我们可以确定模型的每一部分所对应的颜色，从而能够将模型及场景渲染出与输入图片类似的风格。

1.2 本文贡献及后续内容安排

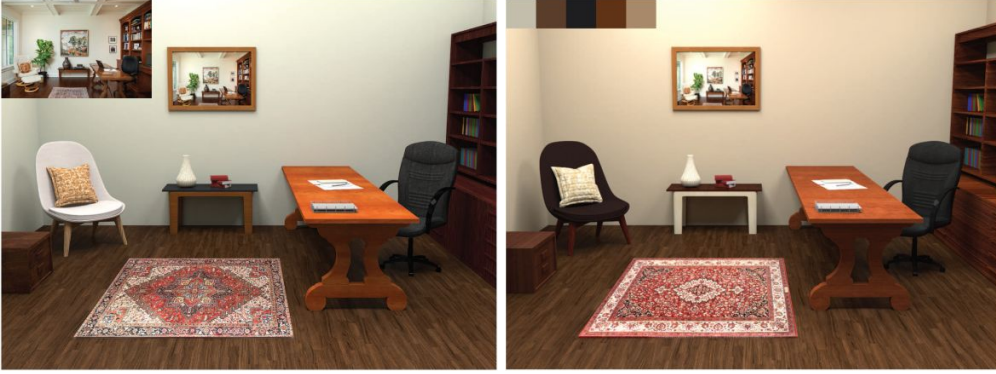


图 1.3: 左：根据一张示例室内场景图片（内部）来对场景自动着色。右：根据用户指定的颜色主题（内部）生成的着色策略

在本文中我们提出一种能自动对室内场景进行着色的算法。首先其自动化是基于一个图片-模型数据库以及学习图片引导的家具模型分割实现的。数据库以层级的形式组织，对于训练图片中属于相同类别的每一个物体，我们详细地用不同的颜色标识物体中的每一个部分并赋予标签，对于每一种类别的家具物体，都有对应的分类器，实现了每种对该物体的不同分割。一旦学习了某种分割之后，我们就能够对任何新的家具模型进行分割并自动地赋予标签，从而进行后续的自动着色流程。

我们的方法提供了两种方法来进行着色，如图1.3所示。对于一个室内场景，如图1.1中的没有材质的三维模型场景，第一种方法是通过在视觉上生成一种与一张图片相似的着色策略。因为我们的方法是面向物体的，因此只要强制要求每一个模型都严格按照图片中相应的物体选择材质或颜色即可。另一种方法是生成与用户提供的颜色主题相近的着色策略。为了实现这一方法，我们针对每一种物体类别，构建了高斯混合模型来对训练图片中的颜色分布进行建模。我们也构建了成对的高斯混合模型来表示两个不同类别的物体的联合颜色分布。对于一个给定的室内场景，我们构建了一个马尔科夫随机场模型来衡量整个场景中每一个物体赋予某一特定颜色主题的概率。该马尔科夫随机场也包含了约束条件来确保对整个场景的着色与用户的输入颜色主题相近。我们使用马尔科夫链蒙特卡洛采样方法来求解优化问题，得到最优的着色策略。

总而言之，我们的方法支持对单个的家具物体着色，对整个室内场景着色，以及通过将给定图的片颜色转移给三维场景进行着色。我们的贡献主要有三个方面。

1. 我们发明了一种家具模型自动着色的算法，允许输入模型是非结构化的，使得用户不必手动地将模型组件和图片中的着色区域对应起来。其中包含了一种图片引导的家具分割方法，训练得到的分类器能够将模型以多种方式进行分割。
2. 我们提供了两种方式进行自动着色：按照示例图片着色，和一种基于马尔科夫随机场的数据驱动框架进行着色。用户调查表明我们的着色效果与真实设计出来的室内设计效果十分接近。
3. 我们贡献了一个结构分明的图片-模型数据库和一个高质量的材质库，前者包括详细的标识信息以及图片和模型之间的对应关系。后者则提供了每一种材质的法线信息、高光信息、漫反射光照信息、透明度信息以及置换信息。

本文的后续内容安排如下：第二章介绍相关工作，包括图片和视频的着色，网格着色和场景装饰，以及室内场景建模和家具摆放等方面；第三章详细介绍本文方法的基础，如何获取图片标识数据以及相关数据库的结构；第四章详细论述模型分割分类器以及着色策略的生成算法；第五章给出详实而完备的实验结果，其中包括一系列的对比试验及用户调查；最后第六章对本文工作加以总结，同时展望后续的可能改进及应用扩展。

第二章 相关工作

本章首先在第一节对着色的相关研究展开了讨论，对一些经典的着色方法进行了介绍；因为我们处理的三维场景，需要对三维网格模型进行处理，为此第二节主要介绍了网格着色的相关算法，以及一些现有的关于室内装饰推荐等方面的研究，并论述了我们方法的不同点与优势；最后一节则介绍了与我们的研究息息相关的室内场景建模方面的研究以及家具布局方面对我们研究的启发。

2.1 图片与视频着色

对于着色一词，起初是用来描述对那些黑白电影进行着色的计算机辅助处理方法，而现在着色则广泛用于计算机图形学中，是指对图片以及视频进行颜色添加的技术。在过去的几年里，已经发展出了很多效果显著的着色技术。

Welsh等人[5]实现了将参考图片中的颜色转移到灰度图中，从而对灰度图进行着色，他们没有通过选择RGB值来给颜色不同的部分着色，而是通过匹配光照和材质信息来将颜色从原图片传递给目标图片，只转移色彩信息而保留目标图片的原始光照信息。同样地，Irony等人[6]提出了一种更新颖的方法，通过分割好的示例图片来对灰度图着色。与依赖于一系列像素级的着色不同，他们尝试对每个像素从更高层次进行处理。首先决定每个像素点应当从示例图片中的哪个分割部分来学习颜色选取，然后通过使用一个邻近匹配矩阵并结合空间进行过滤来给像素点赋予颜色。而在另一篇文章中[7]，用户通过在区域内用画笔划出想要的颜色来表示每个区域的着色策略，随后颜色可以自动地传播到图片中其他的像素上。在对漫画着色[8]和风景画着色[9]的研究中也使用了类似的方法。Sykora等人[10]则针对过去的黑白卡通视频的着色问题构建一种崭新的着色框架，他们将按照示例着色与图像分割问题，基于块的采样和概率推理相结合。Chia等人[11]也通过利用网络上大量的图片建立了一种着色系统，避免了需要用户进行大量交互的麻烦。作为输入，用户只需要提供语义标签和提取图片中主要前景物体的分割线索，而后系统从网上的图片共享网站下载作为参考的图片。Lin等人[12]也基于概率因素图模型提出了一种对二维图案进行着色的方法。该模型基于对示例图案训练并提取其中的风格特征。而在这篇文章[13]中，则研究了颜色移除，即向灰度图以及视频中添加颜色的逆向操作，该方法尝试

保留彩色图片的显著性特征，解决了彩色图片转换成灰度图片时视觉上一些重要的图片特征丢失的问题。

一般的重新上色问题也是和着色相关的，[14]中提出了一种基于示例图片的降低色域维度的方法，同时保留了图片细节。[15]，[16]，[17]都是基于以数据驱动的方式建立的训练集来进行颜色调整。[15]中提出一种增强图片的和谐性同时尽量准确地保留原图片中的信息的方法。[17]则借助深度神经网络来使得图像调整更快更好地实现。

2.2 网格着色与场景装饰

受到图片着色的启发，Leifman和Tal[18]提出了一种对网格进行着色的算法。用户直接在三维模型上划出期望颜色的笔划。而后该算法通过将颜色传递到整个模型上来完成着色。这样的方法对于单个模型着色取得了不错的效果，但并不适用于有多个模型的室内场景，因为在这种情况下，不同模型之间的颜色搭配是十分重要的，会对效果产生直接的影响。另外仅仅要求用户提供一些颜色信息而没有材质信息是不够的，无法渲染出真实照片那样的效果。因此与该方法不同，我们通过一个数据驱动的框架来实现家具模型着色。

在[19]，[20]，[21]中的关于推荐模型材质的研究与我们的研究方向相关。Jain等人[19]提出了一种自动为三维模型赋予材质的方法。然而该方法基于从包含几百个模型和材质的数据库中学习，需要大量的高质量三维模型作为训练数据，因此对于室内场景其应用有限，实际上很难找到大量有着良好几何和材质特征的室内场景作为训练数据。而在[20]中提出的方法则专注于风格转移，将问题描述为材质的组合优化，可以根据图片或视频作为指导，将材质风格传递给目标三维场景中的物体。然而其无法用于需要在一定程度上实现现实仿真的应用，比如室内设计。在[22]中提出了一种方法可以将图片上的材质提取出来并赋予到模型上，将问题分解为为图片中的物体重建基本材质和光照模型。然而这种方法需要单独的图片物体作为输入，因此对于任意的室内场景图片并不适用。[21]中的 magic decorator system能够为三维室内场景自动选择推荐的材质，并将局部材质的选取和全局美学的定义结合在一起。我们的方法与该方法不同于以下一些特征：首先我们的方法能够自动对家具物体进行单独着色。为了使该步骤自动化，基于我们标定的图片-模型数据库，我们提出了一种图片引导的家具分割方法。与此相反的是，[21]中的方法对于一个给定的新的场景没有提供任何机制来给每个单独的家具物体赋予材质，也无法处理非结构化的家具模型构成的场景。其次我们能够将颜色从示例图片转移到室内场景中，如图1.3和

图5.5所示。最后一点，我们的方法是基于家具模型的，而且这一点提升了我们方法的灵活性，因为家具是一个场景的基本元素。就像图5.7中那样，用户可以根据任何一张图片来改变并优化家具着色结果。

事实上目前为止没有任何一个已知工作是为了直观地解决室内物体和场景的着色问题的，室内设计师和游戏开发者通常依赖于一些商业软件，比如3ds Max， Maya等来完成这些交互。所有的这些工具，都需要高度的专业性知识，而且即使对于这方面的专家着色也是一个耗时的任务。

2.3 室内场景建模和家具布局

我们的工作也受到了最近一些关于室内场景建模和家具布局方面研究工作的启发。随着三维模型的获取越来越便利，数据驱动在关于使用RGB-D图片基于语义构建室内场景的相关研究中发挥了重要作用[23]， [24]， [25]， [26]。[23]中提出了一种识别并重建扫描的三维场景的算法。其基于搜索-分类方法在迭代过程中交错地进行分割和分类，通过对近似地重建来获取整个场景的布置。[25]提出一种通过深度图片对室内场景语义建模的交互式方法。用户通过笔划作为分割指导，而后利用分割的语义区域在数据库中检索匹配的三维模型。[26]中建立了一个包含100多个场景和10000多个手动分割和标记的网格模型的场景数据库，基于该数据库重建三维室内场景。

另外[1]， [3]中实现了帮助用户构建三维室内场景，自动模拟家具物体的布置，其结果与人类设计得到的场景十分相似。同样地[2]通过使用室内设计指南进行交互式优化已有的布局。但以上这些研究都专注于建立优化的家具布局，并不适用于场景着色和赋予材质。

第三章 数据库的建立

如前文介绍的，本文所采用的方法是基于数据驱动的，因此首先需要准备大量的训练数据，而且不论是着色所需要的颜色信息，还是分割所需要的多种分割方式的数据，都需要从图片中提取出来。因此如何获取这些数据成为我们实现室内场景自动化着色所需要解决的第一个问题。网络中虽然有着大量的室内设计图片，一些相关研究也将其标定的数据贡献出来，但对于我们的工作而言这些数据还需要进一步的处理和筛选。而我们需要将图片与模型在组件级别对应起来，用于实现图片引导的模型分割，可以直接得到的数据基本上没有达到我们所需要的精细程度。为了解决这一问题，我们首先实现了一个图像标定工具，并通过此工具标记了1680张图片，建立了一个层级式的图片-模型数据库。另外我们也构建了一个材质库用于最终结果的渲染，该材质库中对于每种材质都包含了5种材质贴图，渲染时只需要选定相应的贴图而无图设定相关参数，同时使渲染结果更加真实。本章接下来将介绍我们实验所用到的数据库的构建。

3.1 图片-模型数据库



图 3.1: LabelMe开源在线图片标识工具

3.1.1 图片标定工具

关于图像以及计算机视觉方面的相关研究通常都有着各种各样的标识需要,将图片中标识得到的信息或作为训练数据,或作为基准数据。但手工标识图片这一项工作是十分枯燥乏味的,尤其是对于复杂的场景,而且针对不同的需求对图片标识的功能要求也不同,因此很难构建一个能满足所有标识任务需要的图片标识工具。目前常见的图片标识工具有LabelMe,这是一个开源的在线图片标识工具,用户可以将图片上传到该网站上并使用网页上提供的功能进行图片标识。如图3.1所示,该工具提供了一些简单的操作,可以放大缩小图片,并用多边形工具来选择感兴趣的区域,另外也提供Mask tool来根据颜色自动选择某一块区域。对选好的区域也可以添加一些属性标签进行描述。右侧为标识的物体的标签,可以通过拖动标签来表示表示物体的包含关系。总体上来说该工具能在一定程度上满足我们的需求,但由于要使用该工具首先要把图片上传,且数据在网站上公开的。而我们能设计一个本地的图片标识工具,以及一个图片数据库,自动将标识的数据存入其中。

当然网络上也有一些已有的图片数据库,比如[44]中贡献的室内场景图片数据库,其中包含了67个室内场景类别,并且一共有15620张图片。然而我们的场景仅限于一般的室内场景,比如卧室,客厅等,像该数据库中提供的飞机场,体育馆等室内场景是无法使用的,而且虽然总数很多,排除了一些场景就所剩无几了。而且有些图片中标定的并不准确,或者标定了一些我们不需要的数据,对于一些复杂的场景有些图片标识得过于细致,有些图片分辨率较小,因此里面标识的物体更小,对组件级别而言基本上没有有效的颜色信息。当然这些已有的贡献在很大程度上能加快我们的图片标识进度,但仍需要基于其数据再做进一步过滤优化。

除了我们需要对图片进行标识,提取其中的家具物体的颜色信息和组件信息之外,还需要三维家具模型作为场景的输入以及学习模型分割的材料。网上也有很多三维模型数据库可供下载。另外在准备好模型并已知着色方案后,我们还要选择相应的材质,因为颜色只表示主题,不能直接用纯色给模型上色,要想得到像真实图片一样的结果,需要将颜色映射到材质,使用逼真的材质贴图来使最终的渲染结果接近现实中的场景。因此材质实际上决定了最终的渲染结果,所以我们希望尽可能使用高质量的材质贴图来使得我们的着色效果更加真实。实际上我们设计一种全新的材质表面信息获取方法,通过该方法及相应设备我们可以将现实中的物体的材质提取出来,建立一个包含多种材质的上下左右可拼接的高质量材质贴图的材质库。

3.1.2 图片标识工具介绍

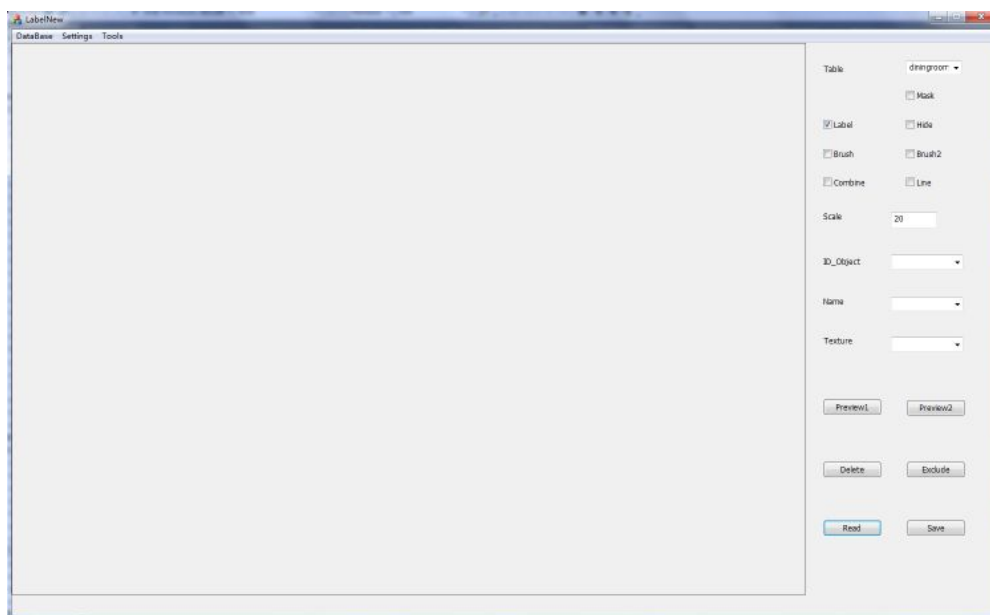


图 3.2: 我们编写的图片标识工具

因此我们仿照LabelMe中提供的功能，并作进一步改进优化，设计了一个图像标定工具。该软件的界面如图3.2所示，该软件使用C++编写，后台数据库使用MySQL实现。该工具的界面大致可以分为三个区域，菜单栏，显示区和功能区，关于该界面的介绍如下：

1. 菜单栏

菜单栏有DataBase,Settings和Tools这三个菜单项，DataBase菜单项负责向数据库中添加表，用户点击子菜单Table后弹出对话框让用户输入新表的名称。Settings菜单项主要用于一些预先设定的配置，包括路径设置，缩放设置，以及预分割设置，保存路径，读取mask所在的路径，以及读取XML所在的路径。因为有的图片数据集是采用XML文件来表示分割信息，而有的则直接用mask图片来表示，因此这里提供了两种方式来读取已有的标定数据。而我们则采用将标定的语义信息保存到数据库中，同时在目录中保存mask图片的方式。最后在Tools菜单项中包含读取XML文件功能，导入功能和导出功能。读取XML和导入功能分别与XML和mask图片作为标定数据的方式相对应，而导出则是将本次图片标定结果保存到数据库中。

2. 显示区

在菜单项的下方，即该软件的大部分区域为图片显示区域，该区域负责显示图片，用户也可以用鼠标在显示区域里结合一些功能选项进行交互。

3. 功能区

(a) 下拉框一共有4个下拉框，最上面的Table代表当前所选择的表，本文所涉及到的场景有客厅，餐厅，卧室，办公室和会议室这五种，因此一共有五个表来存放相应类别的圖片的标识数据。当然可以通过菜单栏中的按钮添加新的表，也会相应的在该下拉框中出现。ID.Object中存放了图片中标定的每一个物体，采用数字前缀加名称后缀的形式，因为一张图片中可能有多个相同类别的物体需要标定，通过前缀的数字来区分它们。Name和Texture则分别表示标定物体的名称和材质，这两个个下拉框中预先存储了一些选项，在实际操作过程中可以直接选择相应的名称和材质，省去了输入的麻烦。

(b) 单选框

界面上提供了多个单选框，用于表示是否启用某些功能。Mask表示是否显示预分割的mask，如果不采用预分割的话这个单选框也不会显示出来。Label表示选择预分割中的某一分割区域，即直接利用预分割的结果。Hide勾选上则会把所有的标记都隐藏掉，显示原图片。Brush1表示使用笔刷工具来进行标记，笔刷为圆形，其半径则由输入框Scale中的值决定，默认为20。Brush2为使用漫水填充算法进行自动分割图片的笔刷，会根据笔划所覆盖的像素点自动选择颜色相近的区域。Line则是使用线框进行区域框选，每一次点击鼠标就会添加一个定点，当线框封闭后即表示区域选择完成。对于笔刷或线框都可以通过鼠标右键来清除或回滚到上一步。如果一开始就用右键进行线框选择则是将选择区域从已有的mask中去掉。Combine则用于将已标定好的两部分组合到一起，用于描述物体及其组件的关系。

(c) 按钮

一共有六个按钮，Read用于读取图片，单击该按钮后会弹出选择图片的对话框。Save则是用于当标记完成一个物体后进行保存，需要注意的是如果标记未完成比如框选是不封闭的则会提示无法保存，如果没有选择相应的物体名称和材质也会有警告弹出。对于组合的物体则可以不填写材质，因为很有可能是由不同材质的部件组成的。两

个Preview按钮分别用于显示mask和原图片的预览图。这里提供mask的预览是为了更明显地将mask表现出来，如图3.3所示，因为可能在修改的过程中会有细小的mask碎片残留，但又不容易直接从显示区域中看出来，通过该Preview直接用黑白两色分别表示mask和背景区域，就很容易发现残留的mask。同样，当点击该按钮后，显示区域也会转变为黑白图，便于修改。而原图的预览则是用于当放大图片只能看到图片中某一部分时也能纵观整幅图片。Delete提供了删除某个物体记录的功能。Exclude也是出于方便图片标识的考虑，当当前标记与之前已有标记发生重叠时，点击该按钮就可以将重叠区域自动从当前标记中去掉，这对于标记紧密连接的部件时十分有帮助。标记完成后会在相应目录下保存相应物体的mask图片，图片名称与序号和物体名称对应，最后再将数据导出到数据库中保存标记的语义信息。

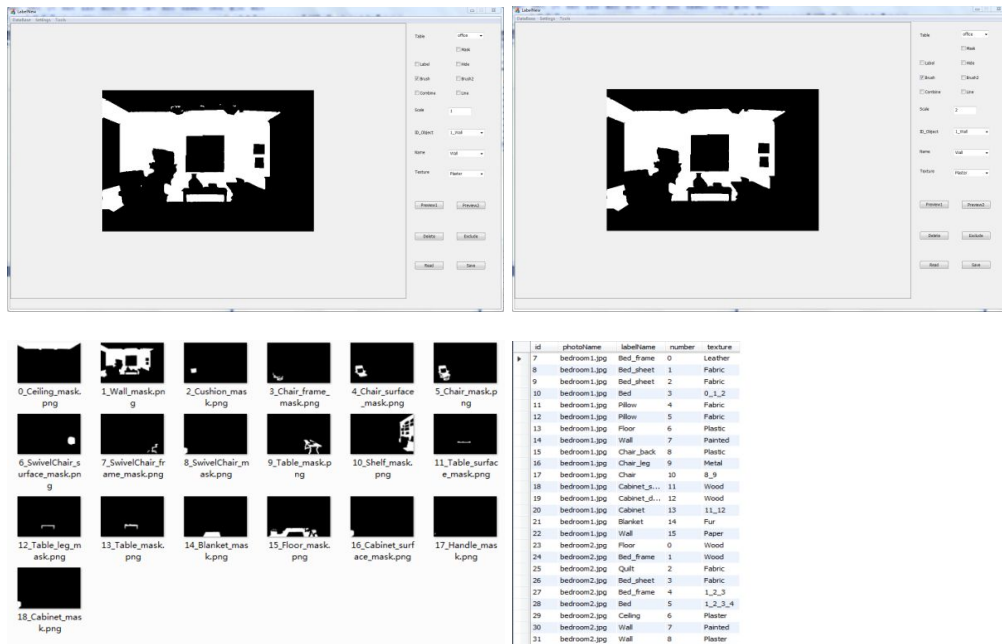


图 3.3: 第一行，左：包含错误的mask。右：修正后的mask。第二行，左：最终保存的图片中物体的mask，右：数据库中的卧室表。

3.1.3 图片标识流程

在读入一张图片后，程序会对读取的图片进行简单的分割操作，该图片分割方法参照了[27]中提出的算法，该算法是一种基于图的分割算法，整张图片可以看作是一个无向图，其中的像素点可以看作图的顶点，每个点与其四邻域的点之间的连线为图的边，这样就够构建了一个图模型。其中每条边都有一个



图 3.4: 图片标识流程

权重，衡量了两个顶点之间的不相似性。实现中就采用了RGB三个通道的像素值的欧氏距离之和。分割问题简单的说就是将所有的像素点分为多个非连通的集合，关键是判断哪些相邻的点应当属于哪个集合。首先我们为所有的点都赋予一个初始的阈值，每个点都可以看作一个集合。然后我们计算出所有边的权重，并按照从小到大排序。从最小的边开始，比对该边的权重是否与两个像素点的阈值小，如果小于这两个点则合并这两个点所表示的集合，并更新阈值为这条边的权重加上之前初始阈值除以当前几何中点的个数所得到的值。合并后的集合则选取rank大的点作为该集合中点的parent，合并次数越多其rank越大。

对所有的边都迭代过后就将整个图片分割完毕了。这样我们就得到了一些预先分割好的图片信息，如图3.4中的(a)所示，我们可以通过选择这些区域而不是像素点来标识图片区域，在这基础上我们再进行修改细化，可以加快图片标识。比如在图3.4(c)中的墙面，我们可以先选取这些分割不太准确的大块的mask，然后再进行细化修改一些边界区域。实际上预先分割好的mask在一些边界区域可以得到很好的效果。

在选取图片中物体的过程中，在一开始根据预分割选取好了大致的mask后还要进行细化，可能会使用到刷子工具，线框工具，反选等等。并根据preview来使标识更加准确。当完成标识后，可以在Name和Texture中选择相应的物体名称和材质。对于需要标识组件的物体，可以先把组件部分标识好，然后使用Combine功能将组件合并成整个物体。最终点击Save按钮将物体保存。当把图片中的所有物体都标识完毕后，就可以点击菜单栏中Tools下的Export来将标记导出到数据库中。同时在每一次保存时，相应的文件夹下都会生成标识得到的mask图片。



图 3.5: 每张图片都标记了三个层次的信息。对于这张图片，第一层次是会议室，第二层次用来表示其中包含的每个家具对象。对于每个对象，第三层次标识了其分割，属于相同类别的家具模型也是按照这种分割方法进行后续处理。

3.1.4 图片-模型数据库结构

我们一共从网络上收集了1680张不同类别场景的图片，包括卧室，客厅，餐厅，办公室和会议室。如图3.5所示，每张图片都在三个层次上进行了标定，第一层次是场景层次，即每张图片都有一个相应的场景名称，包括这五个场景。第二层就是物体层次，每一个家具物体在图片中所占的区域都用一个其所属类别中的标签进行标记，比如沙发，椅子，茶几，会议室桌子，床，地板，墙等等。在第三层次，每个物体区域根据其颜色和材质的不同被划分为多个组件。每个组件也都有一个标签以及材质类别。值得注意的是这个标签并不一定需要是有语义意义的，比如对于一个扶手椅的扶手和椅子腿，如果它们是相同材质和颜色的，那么它们就应当有相同的标签。



图 3.6: 上：椅子模型根据两张类似分割的图片进行标识。下：一个办公室椅子根据另外两张图片的分割进行标识。

我们也收集了200个附带有材质坐标和组件信息的三维家具模型，而且对每个模型构建了与其所属相同类别的图片之间层级式的对应关系。这些模型涵盖了我们需要使用的所有家具类型。对于每种类别中的每一个模型，都要根据相应家具的不同分割方式进行标记，也就是在图片标识中的第三层次中标记的

分割方式（见图3.6）。三维模型设计师通常会为其设计的模型定义清晰的组件，是模型架构清晰便于修改。然而从网上下载模型通常并没有包含完整详细的组件信息。在这种情况下，我们需要搜索模型中的连通区域来恢复组件信息。通常模型中一个有意义的部分会包含超过一个不连通的组件，模型可能会被分割。而我们的模型分割算法可以处理这种情况。

3.2 材质数据库

本节将介绍我们建立材质库所用到的一种材质表面信息获取方法，该方法结合多组拍照技术来提取材质表面的信息的方法，包括得到材质的法线信息、高光信息、漫反射光照信息、透明度信息以及置换信息。由于拍摄的图片质量有限，因此我们提出了光线补偿、图片矫正等预处理算法对图片进行处理，然后采用我们设计的计算方法进行材质表面信息的计算，最后通过我们设计的图像滤波技术、图片拼接等图像处理技术得到上下左右可拼接的高质量材质贴图。

3.2.1 硬件准备

硬件设备即材质表面信息提取所需要的拍摄环境。为了得到更好的用于计算的材质图片，我们设计了一款材质扫描设备，该设备与我们设计的材质表面信息提取算法密切相关，综合计算中所需要的各种图片，该设备包含的硬件设施主要有以下几个部分：遮光箱，透台，相机，LED灯带，滤光膜CPL，单片机。遮光箱是为了避免非拍摄光照对拍摄环境的影响，这里采用了遮光箱进行所有材质图片的采集。透台用于放置待拍摄物体，透台材质为匀光板，可以使从底部照射的光线可以透过匀光板，并起到一定发散光线的作用。灯带保证了在方向光源下进行拍摄，并且尽量减少光线衰减导致的图片质量下降。滤光膜CPL来得到方向光，在所有的灯带上均放置了滤光膜。单片机则用来控制灯光。

3.2.2 图像矫正

在本节中主要介绍了我们对于拍摄图片进行预处理的一些算法。将对拍摄的材质图片进行基本的矫正处理以减少对后续算法造成的影响。主要的矫正算法包括颜色矫正，畸变矫正以及光线矫正。对于光线矫正是必须的，而颜色矫正以及畸变矫正都和我们选择的硬件设备包括相机以及光源有一定的关系，因此可以根据拍摄图片的情况进行有选择的处理。

3.2.2.1 颜色校正

颜色校正也称调色，即对拍摄的图片进行颜色调整。图像采集设备不具有颜色恒常性，往往受到如光源，物体本身反射特性等拍摄环境的影响，使得采集的图像与真实图像的颜色之间存在一定的差异性。我们采用的是白平衡法并结合标准色卡信息进行图像色彩的校正。首先根据拍摄的标准色卡的图片颜色信息与标准色卡进行对比得出色彩偏差曲线，然后对材质图像按照该函数曲线进行色彩调节，最后进行白平衡校正。

3.2.2.2 畸变矫正

由于相机镜头中光学透镜固有的透视失真造成的图像变形即是镜头畸变。随着机械技术的日益精密，相机的制作更加精良，高档的相机利用其本身精良的光学设计以及更为考究的相机材质，通过利用镜片组来优化镜头本身产生的畸变，但是完全消除畸变是不可能的。我们采用了张氏标定算法来进行相机畸变参数的估计。

3.2.2.3 光线矫正

尽管使用了CPL等控制光线的设施，但是光线损失还是无法避免的，因此拍摄的图片还是会存在光线不均的情况，这种情况在设备越大时越明显。本文采用了基于模板方法的线性补偿算法来实现光线补偿，从而减少由于光线损失造成的计算问题。这里所说的模板就是图像模板，由于拍摄期间相机位置以及相机参数都是固定的，因此在进行实际材质拍摄之前首先拍摄一组亮度图片用于之后拍摄材质进行光线补偿。由于在光照方向光线衰减大致呈线性衰减，但线性梯度与材质本身亮度信息有一定的关系因此选择使用图像模板来进行线性补偿。

3.2.3 材质获取

在进行所有的属性计算之前，首先要运行相机自带的矫正程序，比如相机的白平衡矫正，保证相机处于正确的拍摄状态。计算材质表面信息的算法主要通过拍摄材质的两种不同光照下的图片来得到材质表面的几何与物理信息。首先将目标采集样本平整放置于半透明透台中央部分，以标准形式固定。运行软件采集功能，软件控制不同灯光组亮起的同时相机进行拍摄。在相机镜头上放置滤光镜，依次亮起顶部灯组，东部灯组，西部灯组，北部灯组，南部灯组以及底部灯组，打开相机进行拍摄经过矫正算法，依次存储为相应的四张图片，最后除底部灯组外亮起其他灯组，相机进行拍摄存储为对应的图片。由于相机

镜头处使用了滤光镜，因此这组图片滤除了材质的高光信息。然后将相机镜头处的滤光镜移除，依次亮起顶部灯组，东部灯组，西部灯组，北部灯组，南部灯组，打开相机进行拍摄，经过矫正算法依次存储为相应的四张图片，最后除底部灯组外亮起其他灯组相机进行拍摄存储为对应的图片。我们为了保证图片的质量，将所有拍摄以及计算的图片都存储为以.tif为后缀的图片。根据这些图片，我们可以获取相应的材质表面信息包括材质表面的法线信息，漫反射信息，高光信息，透明度信息以及置换信息。

3.2.3.1 材质法线信息获取

法线计算有两种方法，一种是直接使用遮光拍摄架固定在东西南北的四组灯光组拍摄的图片通过光线补偿进行计算，称为补偿计算法，另一种方法是使用滑轨扫描得到东西南北四方向光线无衰减的图片计算，称为直接计算法。直接计算法不需要对拍摄的图片进行光线补偿，在这种方法中我们改进了硬件设施，拍摄箱相比于补偿计算法多了两组导轨，灯带固定在导轨上，使用电脑控制导轨的速度以及相机的曝光时间，使得在相机曝光时间内导轨从材质的一侧滑动到材质的另一侧，这样就可以避免因距离关系导致拍摄图片受到光线衰减的影响，因此最终拍摄出来的图片光线均匀，然后再通过补偿计算法得到法线信息。

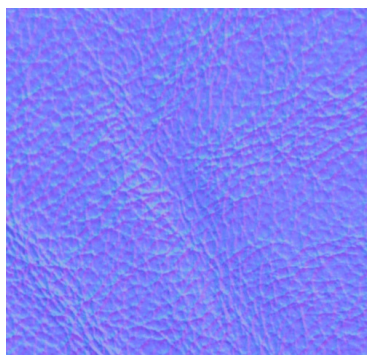


图 3.7: 材质法线信息图片。

3.2.3.2 材质漫反射信息获取

材质的漫反射（颜色）信息反映了材质在光照情况下各个方向的反射情况，因此可以说是材质信息最直接的光线体现。材质的漫反射信息不应该存在高光信息，因此在我们的设备中使用了滤光膜CPL来滤除材质的镜面反射。通过顶部灯光照射所拍摄的图片大致可以看做材质的漫反射信息，但是考虑材质的凹凸属性，仅仅使用顶部灯光并不能完美的体现材质的颜色信息，我们使用了全光照下拍摄的图片，即全光照图来进行调整，这样减少了因材质凹凸所造成的

颜色信息的丢失。通过将这两张图片叠加就得到漫反射信息。

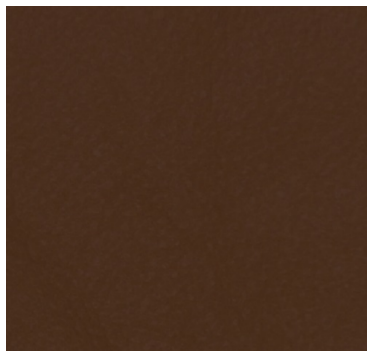


图 3.8: 材质漫反射信息图片。

3.2.3.3 材质高光信息获取

材质的高光信息反映了材质的镜面反射情况，不同材质的镜面反射情况不同，金属材质的镜面反射率大，因此对于金属材质得到的高光信息亮度就大，而对于反射率小的布料等材质高光信息亮度就小。鉴于不同的材质高光颜色、亮度等都不同，因此本文采用了滤光膜CPL来得到材质的高光信息。我们在第



图 3.9: 材质高光信息图片。

一组拍摄的图片中，由于镜头带有CPL相当于拍摄的图片并不带有高光信息，而第二组拍摄的图片中镜头处移除了CPL，没有滤除掉高光信息，因此可以使用两组图片中顶部灯组拍摄的图片进行高光信息的计算。

3.2.3.4 材质置换信息获取

置换贴图的使用通常是让点的位置沿面法线移动一个贴图中定义的距离。它使得贴图具备了表现细节和深度的能力，且可以同时允许自我遮盖，自我投影和呈现边缘轮廓。置换信息主要作用就是按置换图中红色通道的灰度值，控制原图像素水平方向的移动。按绿色通道中的灰度值，控制原图像素垂直方向

的移动。蓝色通道不参与置换。我们采用了一种简单的算法计算置换贴图，这里主要用到了顶部拍摄图片和全光照图片，将两张图片每个像素求平均值，得到的新的图片转换为灰度图，最后对灰度图进行调整得到置换信息。

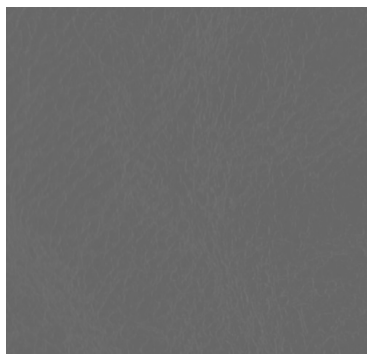


图 3.10: 材质置换信息图片。

3.2.3.5 材质透明度信息获取

材质的透明度信息主要体现了材质的透光属性，我们将这种属性可视化为图片就得到了材质的透明度信息贴图。材质的透明度信息主要通过遮光箱底部灯组拍摄的图片计算得到。当底部灯组亮起时，如果材质透光性好，则拍摄的

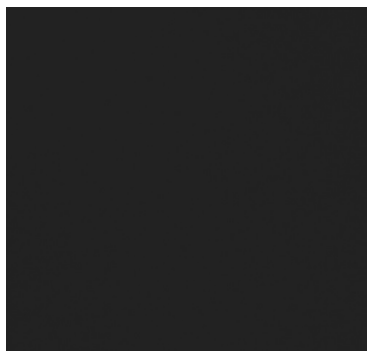


图 3.11: 材质透明度信息图片。

图片亮度较大，如果材质是不透光的，则拍摄的图片应该是暗的，如果完全不透光拍摄的图片应该是黑色的，因此可以根据底部灯组拍摄的图片的亮度确定材质的透光属性。

材质贴图对于照片级的真实感渲染很重要。直接从图片中提取的区域可能因为透视和光照等原因不足以用于渲染。因为时间和材料的原因，材质库还在构建中，因此为了生成最终的实验结果我们先通过在网上收集了2650个材质来补充现有材质数据库。虽然这些材质贴图只是单纯的贴图，不包含法线高光等信息，但我们通过在3ds Max中进行一些调整也足够用于渲染出比较真实的效

果。日后我们会用新建立的材质库替换现有的材质库，生成新的室内场景着色结果。在图片标识的过程中，每一个物体或组件都被赋予一个材质类别标签，即这些材质与图片标识的第三层次相关联。下图是材质图片截图。

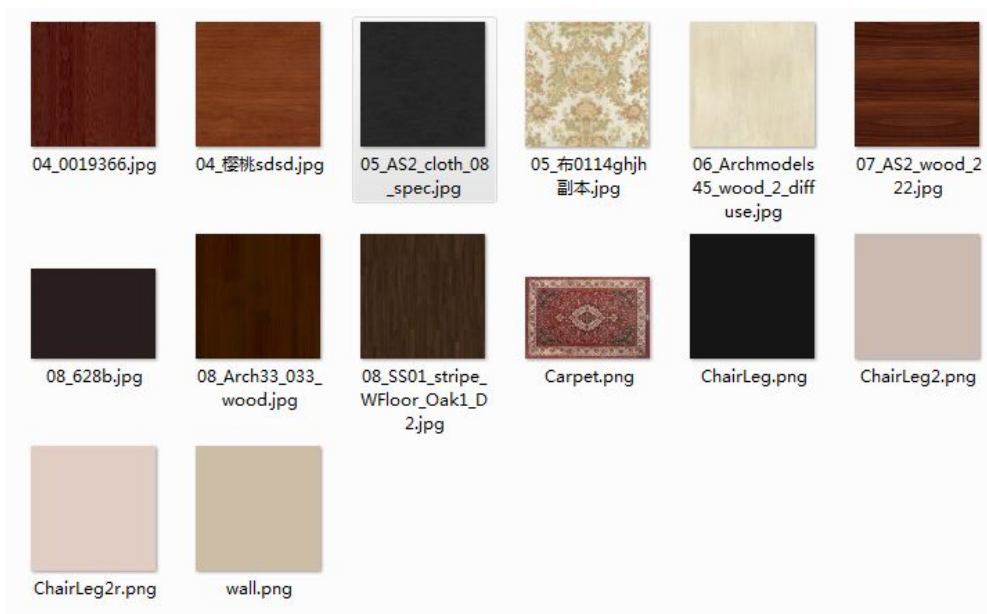


图 3.12: 对一个办公室场景着色所选择的材质。

第四章 基于数据驱动的着色算法详述

本章将介绍本文内容的核心即基于数据驱动的着色算法。着色问题的输入为一个三维室内场景，通过已知类别的家具模型构建。我们最终目的是为每个家具模型以及整个场景提供合适的着色方案。我们的方法是基于数据驱动的，通过网络获取的大量专业摄影师拍摄的或经验丰富的室内设计师设计的室内场景图片进行学习来实现着色策略的推荐。这个框架主要可分为两个关键部分：对每个家具物体着色和对整个场景着色。

对单个物体着色。给定一个家具模型和一张作为参考的家具图片，我们希望能够严格按照图片对模型进行着色。一个家具模型通常由一组拓扑不连续的组件组成，因此室内设计师可以手动地从当中选出一部分来赋予一种材质。而我们的做法关键是要以图片作为参考，根据这些包含不同颜色的区域自动地从模型中选出那些应当赋予一致颜色的组件部分。比如在图3.12中的椅子包含两种主要颜色灰色和黑色。如何根据这张椅子图片自动地为椅子模型着色是个难题。这可以归类为由家具图片引导的模型分割问题。我们通过构建一个在组件层次详细标记了模型以及相应图片之间的相互关系，并学习图片引导的模型分割来将这一过程自动化。根据分割结果，我们就可以容易地用与示例图片相似的材质来对模型进行着色，这是通过在材质数据库中搜索与模型图片的颜色最相近的材质实现的。这一步即图中流程所示，将在下一节详细介绍。

对整个场景着色。基于上述家具着色方法，我们可以很容易地将一个场景着色为与一张图片相似的颜色。换言之，我们能够按照用户的喜好来推荐着色策略。对于整个场景而言，不同家具物体之间的颜色组合对于创建一个和谐、让人感到舒适的室内设计是很重要的。我们使用一个马尔科夫随机场（MRF）模型来衡量所有家具选择某一颜色的概率以及不同物体之间选择不同颜色组合的概率。我们构建了衡量一个模型在相应物体类别中的颜色主题分布概率的高斯混合模型（GMM），对每个家具模型的颜色推荐是通过对高斯混合模型的密度方程采样得到，同样地，颜色组合的状态空间也是通过高斯混合模型来构建的。人们通常会对特定的一些颜色有各自的偏好，并倾向于选择这些颜色来装饰场景。为了满足这一要求，我们让用户提供一个目标颜色主题，作为我们的MRF模型的约束条件。我们采用马尔科夫链蒙特卡洛采样算法来求解优化问题，最终生成使整个场景看起来自然美观且符合用于需求的颜色主题。我们将在场景着色一节详细介绍。

图4.1展示了我们着色框架的流程。



图 4.1: 三维模型和场景着色框架流程图

4.1 图片引导的家具着色

对于给定的一个任意家具模型和一张从我们数据库中选择家具图片，我们的目的是对模型进行着色使得其颜色与参考图片风格一致。家具模型通常是由多个不连通的组件组成的。然而图片中的家具通常有多个着色区域。因此关键是要建立这些模型组件与图片中不同颜色区域之间对应关系，从而使得不同的组件部分能够按照不同的家具图片区域来着色。这可以看作一个对这些组件部分的分类问题，或更进一步是一个为了给每个组件部分赋予标签的分类问题。这些标签不一定是具有严格意义的，它们只是由图片中不同颜色的区域来决定的。事实上因为我们数据库中的图片是已经经过标识的，里面包含了相应物体的颜色信息和分割信息，因此我们可以首先按照图片的分割方式来对室内三维模型物体进行分割，而后根据图片中的颜色信息选取相应的材质赋予三维模型，来完成三维家具物体的着色。

4.1.1 图片引导的家具分割

尽管网格分割问题已经得到了广泛研究，之前的方法都是在基本的层次上即为每个单独的三角形网格来赋予标签[28], [29]。于此不同的是，我们面临的问题是基于组件的网格分割，因为三维家具模型都是由一系列拓扑不连通的组件组成，每一个组件则是由三角形网格组成。而且我们的分割是基于家具图片引导的，也和传统的基于组成部分的或语义的分割不同[30], [31]。至此我们提出基于网格级和组件级的特征来学习家具模型分割。

对于一个给定的家具模型，我们首先对每个三角形网格计算一组局部特征，包括曲率，主成分分析特征（PCA），形状直径函数特征（SDF），平均测地线距离特征（AGD），形状脉络特征（Shape Context）和旋转图像特征（Spin Image），再以单个三角形的面积作为权重，通过所有三角形的面积之和进行归一化。即

$$F = \frac{s}{sum} f \quad (4.1)$$

f 为计算出的一种特征， s 为该三角形的面积， sum 为所有三角形的面积之和， F 为归一化后的特征，将上述这些特征组成一个特征向量得到最终的局部特征。这些特征的详细介绍可以参考[31]。我们计算这些特征并将其拼接到一起，组成了一个三百多维的特征。

我们也对每个组件计算了一组全局特征。使用高斯曲率、SDF特征和AGD特征对每个组件提取全局特征。以高斯曲率为例，首先对每个组件的所有三角形计算高斯曲率，然后计算其加权平均值，中位数，方差，偏斜度，峰度，和高斯曲率的直方图一起组成特征向量作为衡量该组件的全局特征，其中加权平均值用三角形面积加权。对另外两个特征也用该方法进行计算。

与[31]类似，我们使用JointBoost进行训练。JointBoost是一种boosting分类器，因此其具有boosting分类器的特点，能够为不同的特征赋予不同的权重，可以有效地处理大规模的输入特征，并自动进行特征选取，事实上我们也并不确定哪些特征对模型分割的准确性影响较大，所以我们完全交给分类器选择，对于我们这种特征维度较高的情况比较适用。另外JointBoost可以进行快速的序贯学习，因为其可以在不同的类别之间共享特征，对于不同类别的特征空间有重叠的情况，可以减小分类误差。事实上，随着分类类别的增加，为了取得同样准确的分类结果，对于JointBoost所需要的特征数呈对数级增长，因此相对来说取得同样的分类准确度其所需要的特征数也较少。[35]中详细介绍了JointBoost。

我们接下来将模型输入分类器进行分割，即确定模型中每个组件的标签。对于输入模型，我们为每个三角形网格训练了一个局部分类器 J_l ，为每一个组件 P 训练了一个全局分类器 J_g 。 P 被赋予标签 l 的概率表示为，

$$p(P, l) = \sum_i \frac{1}{Z} area(m_i) * J_l(m_i) + \lambda J_g(P) \quad (4.2)$$

这里 m_i 代表组件中的第 i 个三角形， $area(m_i)$ 表示其面积。 Z ，作为归一化参数，是组件中所有面片的面积之和。 λ 这里是一个协调因子，用来平衡局部和全局分类器，最终根据所有训练模型的ground truth优化为0.4。我们最终将概率

最高的标签赋予组件。因为我们所处理的模型都是由一系列拓扑不连续的组件组成而非网格为单位，因此相比于直接对网格进行分割可以取得更高的准确度。

4.1.2 家具模型着色

根据分割结果，我们可以轻松地选取相似的材料为模型着色，使其与参考图片相似。虽然我们已经拥有参考图片，但我们也不会直接选取图片中的颜色，因为通常图片组件部分都很小，且因为图片质量等问题都比较模糊，所以我们需要一种抽象的方式来表示颜色。最终我们采用颜色主题来描述图片区域的颜色。仿照前人关于颜色兼容和增强的研究[36], [37]，我们将颜色主题表示为由五个颜色组成的调色板。这样的一种表示方式广泛地为艺术家和研究者使用，可以在视觉上直观有效地表示颜色主题。另外其计算开销也相对较少。为了从一张图片中提取出相应的调色板，我们对像素点的颜色进行K-means聚类。这产生了一个带有K个箱子的颜色直方图，而其中最大的5个箱子对应的颜色就构成了调色板。我们根据经验将K设为50。

对于一个对象的组件部分，我们按照如下的距离度量方程从相关的材质类别中选择出M个最相似的材料，

$$D(C_O, C_T) = \sum_{k=1}^5 \min_j \|C_{Ok} - C_{Tj}\|_2 \quad (4.3)$$

这里 C_O 和 C_T 分别表示一个物体部分和我们材质数据库中材质的颜色主题。 C_{Ok} 为 C_O 中的第k项，而 C_{Tj} 表示 C_T 中的第j项。我们根据家具模型的分割结果为其赋予最相似的材料，并为用户保留剩下的M个材质样本，以便于其按照个人喜好进行选择。作为折衷的策略，我们随机地从M个材质样本中选取一个作为最相似的。在我们的实现过程中M设为10。

其他重要的物理参数，尤其是反射率，可以很容易地从已有的材质数据库中获取，因为对于特定的一种家具只有少量类型的材料是常用的。而实际上如果我们使用新的材料库，其中包含高光贴图，也就无需设置反射率。

4.2 场景着色

接下来我们介绍针对场景的着色，除了输入一个从我们数据库中选择出来的已知类别的三维模型组成的一个室内三维场景，还分为两种情况：按照示例图片进行着色和按照前文介绍的颜色主题进行着色。示例图片同样也是从我们的数据库中选择，因为需要标识信息，而颜色主题则对应了期望目标场景的着色结果。

4.2.1 按照示例图片着色

在这种情况下，对于输入的一个三维场景，其中的家具类型与用户输入的希望作为着色参考的图片中的家具类型一致。这样的场景可以按照上文所介绍的图片引导的着色策略轻松地场景中的每个物体进行着色，即将每个三维模型都与图片中标识的每个物体对应起来，采用依照参考图片对单个三维模型物体着色的方法进行着色。

4.2.2 基于数据驱动的着色框架

从现在开始，我们将详细介绍基于数据驱动的三维室内场景着色的算法框架。这种情况与参照图片不同，虽然参考图片也并非直接从图片中提取颜色，但至少比较直观。而这里我们除了一个已知物体类别的三维室内场景之外，还需要用户选定一个目标颜色主题，而非之前所说的参考图片。颜色主题相比于参考图片而言则抽象了很多，虽然其表示了整个场景的颜色风格。该算法的目的是为整个场景给出推荐的着色策略，使得场景中的每一个模型的着色都自然而且视觉上美观，并使得整个场景的颜色搭配与用户输入的一致。因此首先我们的思路首先是根据输入的颜色主题确定每个物体的颜色主题，使得这些颜色主题符合我们之前的要求，然后根据颜色主题在我们的图片-模型数据库中找到拥有最相近颜色主题的图片，由此就拥有了着色和模型分割的参照，也就转化为对单个家具模型着色的问题。

我们根据问题构建出一个马尔科夫随机场模型。如图4.2所示，假设我们的输入图片中包含五个物体：墙，桌子，椅子，地毯，地板，场景中的每个三维家具模型表示为该图模型中的一个节点，而属于相同家具类别的每一张物体图片都是相应节点的一种可能状态。对于两个家具物体，只有当它们同时出现在数据库中的图片里时，在图模型中才会有一条边连接这两个节点。为每个模型赋予一个颜色主题，也可以看做是赋予一个标签，我们应当使得每个相同标签内部的变化均匀具有连续性，但在不同的标签边界处要保持强烈的不连续性。对于这样一个问题通常都会转为能量优化问题。

由此我们根据该马尔科夫随机场模型构建相应的能量方程，其定义如下，

$$E = E_D + \beta E_S + \gamma E_C \quad (4.4)$$

通常的能量方程只包含前两项，而这里我们还添加了一个约束项。这里 E_D ， E_S 和 E_C 分别表示数据项，平滑项，和表示用户对于整个场景所倾向的颜色风格的约束项。 β 和 γ 则用来平衡这三项的权重系数。为了强调用户所倾向的颜色风格的重要性，我们分别将其设为1和10并在实验中保持不变。对于每个物体类

别，我们基于训练图片中的该类物体使用高斯混合模型对颜色主题进行了建模。回顾上文所说的，一个颜色主题是通过由五个颜色组成的调色板表示的，因此对于RGB颜色空间，一个颜色主题据相当于一个15维的向量。相比于特征空间而言，一个高斯混合模型所表示的分布太小而显得有些不足，为了解决这一问题，在实现中我们采用拥有16核的5个高斯混合模型，其中的每一个代表了颜色主题中一项的分布，当求取一个物体的颜色概率时我们将这5个高斯混合模型累加起来，由此我们拥有了数据库中不同类别的物体的颜色对应的概率分布。

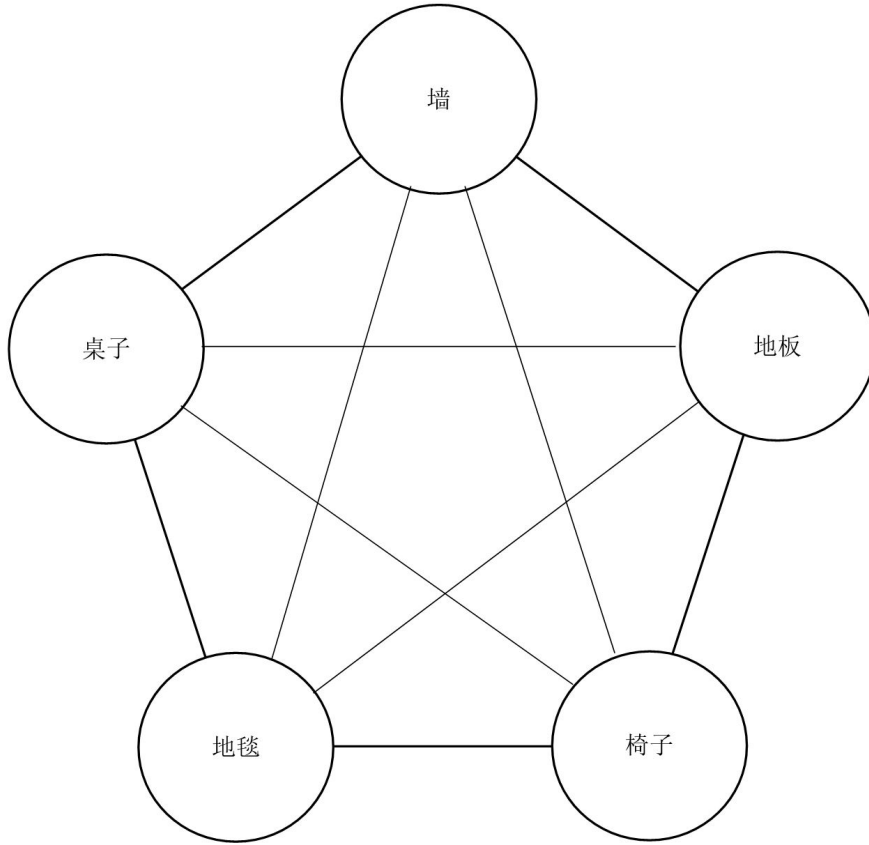


图 4.2: 根据场景构建的无向图模型

数据项 E_D 的值为所有模型选取对应颜色主题的概率值之和。该项衡量了某一着色策略对一个家具模型而言是否在我们的数据库中即现实中是常见且美观的。一个少见的颜色策略会拥有很小的概率值。其按照如下公式计算，

$$E_D = \sum_i \log(G(C_{Mi})) \quad (4.5)$$

这里 C_{Mi} 表示每个单独的家具模型的颜色主题。我们将 C_{Mi} 的颜色主题在高斯混合模型中对应的概率值表示为 $G(C_{Mi})$ 。

第二项 E_s 可以看作是一个平滑项，其表述为，

$$E_s = \sum_{ij} \log(G(C_{Mi}, C_{Mj})) \quad (4.6)$$

平滑项一般表示两个相邻节点之间选取不同标签的概率。在这里该平滑项是通过一个二元高斯混合模型计算得到的，表示两种物体的颜色主题的联合分布。输入为两个物体的颜色主题，衡量了场景中两个物体的某种颜色组合是否是常见的。在实现中我们同样地使用8核的高斯混合模型，来衡量在同一幅图片中出现的两个不同物体之间的颜色主题之间的相互关系。

第三项 E_C 代表了用户约束，确保了整个场景的着色结果看起来与用户输入的颜色主题 C_U 相近。对于每个家具模型，都对其颜色主题中的每一项计算与目标颜色主题的差别，并累加起来。

$$E_C = -\frac{1}{Z} \sum_i \sum_{k=1}^5 \min_j \|C_{Uk} - C_{Mij}\|_2 \quad (4.7)$$

这里 Z 是归一化因子，使得第三项与第一项有着同样的量级。

我们的目的是使上述能量方程最大化。这是一个难以求解的高维度，非凸组合优化问题。马尔科夫链蒙特卡洛（MCMC）方法常用于从多维分布中进行采样[38], [39]，尤其是高维度的问题。我们因此采用马尔科夫链蒙特卡洛采样法来对整个场景求解一个最优的颜色配置。马尔科夫链蒙特卡洛是基于构建一个马尔科夫链来实现的，该马尔科夫链因其均衡分布的特点保持了良好的分布特征。优化的过程中通过不断迭代，针对处于当前状态的变量，对于下次迭代生成一个候选状态。通过选取合适的状态转移方程，当经过一定次数的采样后会达到收敛状态，能量方程的值也将稳定下来，即得到最终的优化结束。我们使用了一个名叫UGM[40]的工具包来实现马尔科夫链蒙特卡洛采样，该工具可以解决多种无向图的相关问题。目前使得公式4.3最大化的着色策略将作为推荐的着色策略，我们在得到了场景中的每个物体所对应的颜色主题后，再从数据库中选择相应的物体图片，采用在上一节中介绍的方法来完成对模型进行分割以及材质的选取。最终根据颜色主题对一个会议室场景着色的结果如图4.3中的(a)所示。

然而用户可以根据其喜好从迭代过程中的多个采样结果中选取着色策略，而非唯一的最终结果。因为中间结果都具有一定的随机性，也有可能产生不错效果。如图4.3中的（b）和（c）所示，即使这两个着色结果并不是使能量方程最优的着色方案。

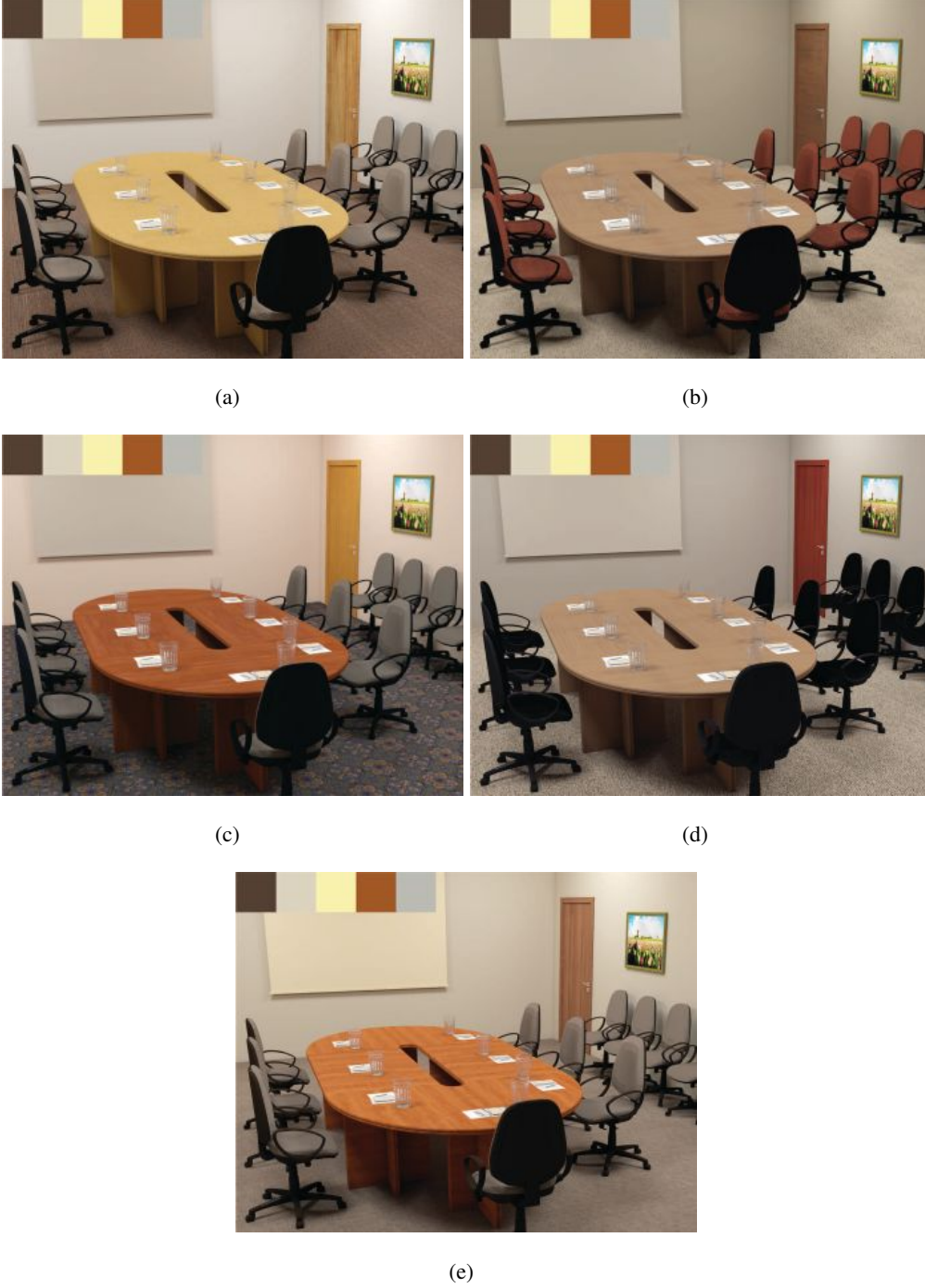


图 4.3: (a): MCMC优化得到的结果。(b) 和 (c): 在MCMC迭代过程中得到的其他采样结果。(d): 忽略用户约束条件得到的结果。(e): 只考虑用户约束条件得到的结果。

另外我们也衡量了用户约束项 E_C 的作用。如图4.3中的(d)所示的, 去掉这一项, 整个场景与目标颜色主题相去甚远。作为对比, 如果只考虑用户约束项, 则使得深棕色的桌子与周围物体颜色格格不入(图4.3(e)), 因此整个场景看起来不如图4.3(a)根据最终优化的到的结果和谐美观。

因为我们的颜色选取是通过求解能量方程得到的, 最终材质选取和模型分割都是通过从我们已有的数据库中找到的参考图片来指导完成的, 实现了整个颜色选取流程的自动化。而最终的渲染因为涉及到灯光以及视角的选取, 还是需要用户进行交互来实现。

第五章 实验应用以及用户调查

5.1 网格分割

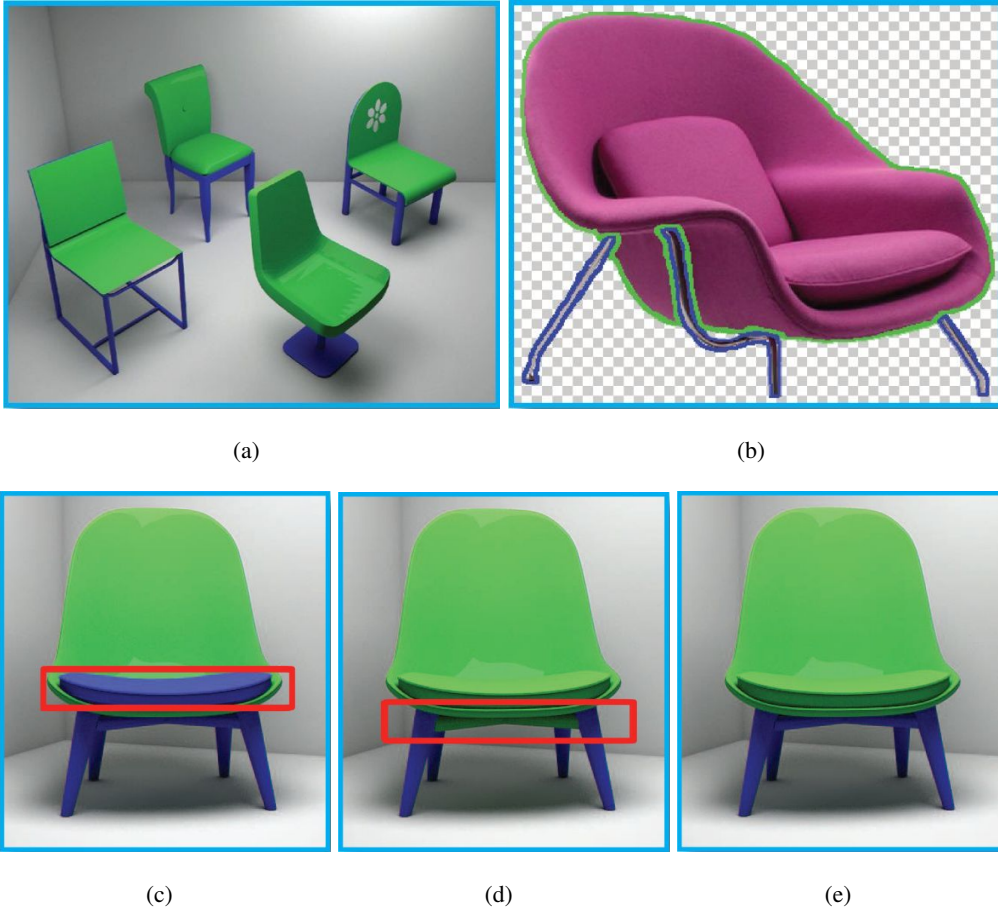


图 5.1: 图片引导的网格分割结果。三维模型图片中的红色矩形区域表示只使用局部分类器得到的错误的分割结果（第三张图）。同样的，只使用全局分类器得到的结果也会出现错误（第四张图）。通过将两者结合，我们得到与示例图片一致的分割结果。

网格分割是我们框架中的关键一步，我们将首先验证其有效项。图5.1展示了一个关于我们的图片引导的模型分割例子。值得注意的是如果只选用局部或全局分类器，图中红色三角形区域将会出现分割错误。通过结合局部和全局特征的分类器，能够得到更好的分割结果。

我们更近一步量化地衡量分割结果。除了准确地测量分割是否准确，我们

还采用了随机索引和一致性误差来衡量分割的效果[41]。前者衡量了分割结果和实际正确分割结果之间每一对面片的标签一致性，而后者量化地衡量了它们的层级相似性，包括全局一致性误差(GCE)和局部一致性误差(LCE)。我们做了一些修改来使得它们与我们的组件级别分割相适应。我们从模型数据库选取了39个模型作为训练集，剩余的用于测试。为了与其他代表非相似性而不是相似性的度量标准相适应，我们同样计算1-precision而非precision。如图5.5所示，我们所采用的结合了局部和全局特征的分割方法相比于使用局部特征作为训练特征的方法取得了很明显的优化效果。

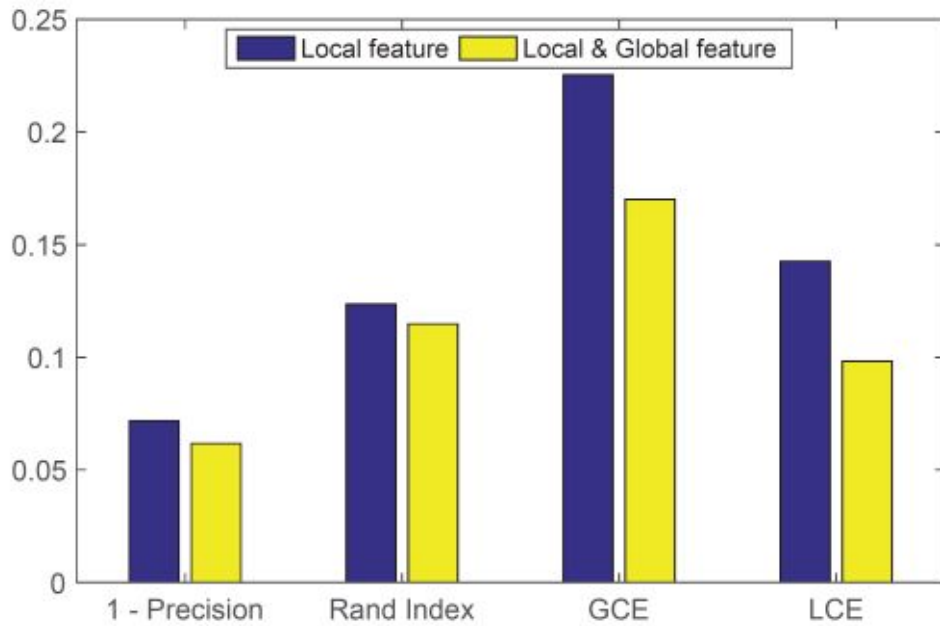


图 5.2: 对我们的网格分割算法结果统计表明通过结合局部和全局几何特征相比只使用局部特征有很大的提升。

5.2 家具和场景着色

我们针对典型的家具模型和室内场景对我们的方法进行了测试。所有的关于家具和场景着色的结果都是通过我们的方法自动生成出来的。输入到该系统的是场景中的三维模型，并附带其类别信息，即该物体是桌子，沙发或者床等等，和场景的类别，如客厅，办公室，会议室等等。表5.1显示了关于我们的场景的统计数据，包括物体的数量，模型的组件数量，以及用于着色的材质样本数量。

Scenes	Office 图1.3	Meeting Room 图5.5 2/3行	Living Room 图5.5 2/3行	Dining Room 图5.5 2/3行	Dining Room 图5.6	Living Room 图5.6	Meeting Room 图5.6	Bedroom 图5.7
Objects	10	13/13	14/14	14/12	17	11	18	7
Components	43	147/309	91/53	162/93	97	69	258	57
Materials	13	8/8	16/15	11/10	11	11	8	11

表 5.1: 本文中用到的场景的统计数据

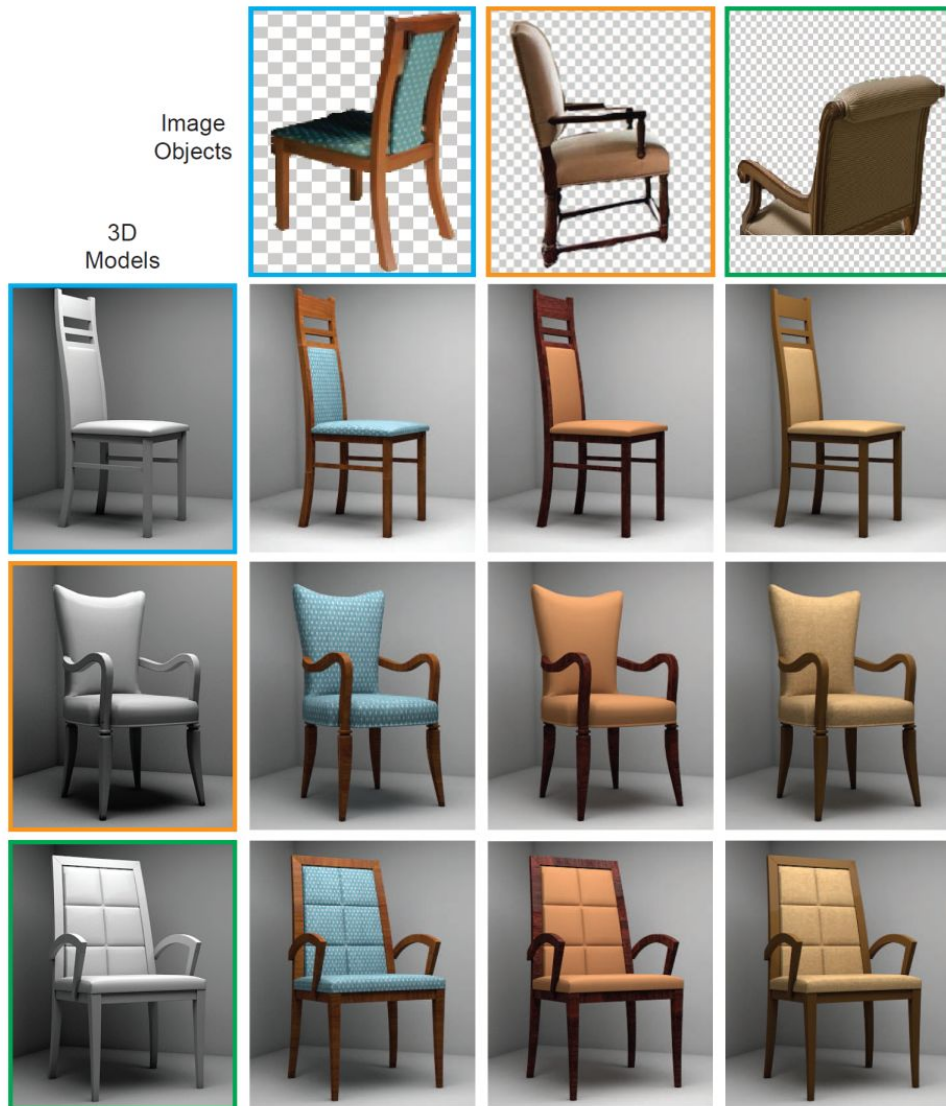


图 5.3: 使用三张不同的参考图片的得到的三个不同椅子（扶手椅）模型的着色结果。

5.2.1 家具着色

图5.3显示了我们以三把扶手椅图片作为参考，对于三个扶手椅模型的着色结果。在我们的数据库中，用作训练的所有的扶手椅都被标记为“chair”。在这个例子中，我们将相同的分类器用于这三个不同风格的椅子模型进行分割。我们的方法所生成的着色策略严格遵守三张训练图片，即使三个模型的分割结果并不相同。最终生成的9个着色后的椅子看起来显得十分自然和谐。

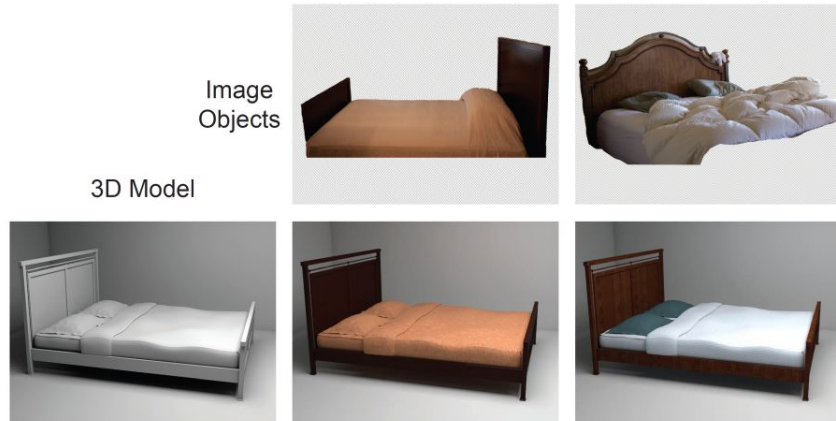


图 5.4: 使用两张示例图片作为参考对一个床模型进行着色的结果。

图5.4显示了参考两张床的图片对一个床模型进行着色的结果。值得注意的是在第二列，床模型被分割成了两部分，分别对应参考图片中的床架和床垫。而在第三列中，该床模型被分割成了三部分，通过另一个分类器将两个枕头也分割了出来。事实上在我们所有的结果中，对于每一个床模型，枕头和床垫都是属于一个统一的三维模型，并且这些模型都有着很多互不相连的组件。在我们的图片-模型数据库中，属于床的枕头和床垫并没有被标注为独立的物体。然而我们的方法能够自动地为床，枕头和床垫自动赋予不同的颜色。

5.2.2 场景着色

基于物体的特征为我们的方法赋予了能够按照示例进行着色的特点。即我们能够通过从用户提供的参考示例，也就是一张示例图片进行颜色转移来对输入的场景着色。我们将用图5.5中的多个结果来展示该特点。图中的第一行显示了示例图片，第二行显示了相应的结果，其中的家具是参考示例图片中的家具位置进行摆放的。在第三行，我们展示了采用不同家具模型和摆放的结果。

图5.6首先首先展示了根据一个用户输入的颜色主题所生成的一个餐厅着色推荐结果，输入的颜色主题在图片当中。该餐厅由一个桌子，六把椅子，一个圆桌，一个欧式沙发以及上面的抱枕组成。地毯，窗帘和墙也是自动着色的，

因为这些物体在图片标识过程中都有各自独立的标签。餐具，花瓶还有两幅风景画是由用户自己摆放的。接着是一个客厅场景，包括一个电视柜，一个橱柜，一个茶几，一个由两部分组成的棕色布艺沙发，一个两座的布艺沙发，包括两个垫子，和一个小桌子。其中木质地板也是自动着色的。最后是一个典型的会议室。



图 5.5: 我们的方法一个特别之处在于能够参照示例图片（第一行）对三维场景（第二、三行）进行着色。



图 5.6: 以用户输入的颜色主题（每张图片内部）作为约束的着色结果。

交互式优化。用户可能会想要根据个人喜好来对场景进行优化，比如，修改特定某个家具的颜色。我们的框架也支持这一点，因为我们的方法是面向物

体的。对于给定的用户选择的颜色主题作为输入，图5.7首先展示了一个典型的卧室的结果。在下一步，用户按照一张卧室图片的颜色修改了床的颜色，从深红色修改成了棕色。我们的系统为大部分家具模型推荐新的颜色配置来使得整个场景与修改后的床相适应。接着用户又将纤维材质的地毯修改成了木质地板，为了匹配这一修改，柜子也调整成了更亮的颜色。这样一来，既能够保持整个场景的颜色对比，也保证了颜色组合的和谐。



图 5.7: 用户通过交互来调整一张卧室图片结果（左）。首先用户选择一张卧室图片中的床（中间图片左上角的图片）作为参考来修改床的颜色，场景自动更新为中间图片的结果。用户接着又将一开始的地毯修改为木质地板，得到新的着色结果（右）。

5.2.3 实用性

对于我们之前所展示的所有实验结果，参照的图片都是从在训练阶段预先分割好的图片中选取的。但是如果用户对训练集中的图片都不满意，则需要一张新的示例图片。在这种情况下，如果通过新的图片引导的模型分割结果与我们预先学习的分割是不同的，则需要手动地选择相应的分类器。另外，我们可能需要手动地分割新图片，并重新学习得到一个新的分类器。

然而我们的数据集目前包含大约2000张图片，涵盖了各种不同的家具，包括各种不同的分割情况。我们在Google Image上按照关于家具的多个关键字搜索了多张图片来测试我们系统的实用性。图5.9显示了一张关于椅子搜索结果的截图，这些搜索结果可能作为用于输入系统的图片。其中的大多数都属于两种类型的椅子。其中一般的椅子标记为蓝色，转椅标记为绿色。我们逐个测试了这些图片，这些图片对应的分割方式都可以在我们预先训练好的分割分类器中找到。标记为黄色的图片是与椅子无关的。这些通常可以归类为沙发，实际上在我们的数据库中也是被标记为沙发而非椅子。其他的图片则标为红色，其中的每一个都只包含一种材质。因此如果选取其中的任何一张作为示例，都不需要进行分割。

这表明在大多数情况下，用户并不需要训练新的分类器用于分割，即使用

户倾向于选择不属于我们数据库中的图片。

局限。我们需要手动地去修正错误的分割结果，这是我们方法的一个不足之处。然而，就如上一节所提到的，我们训练的分割分类器涵盖了大多数的分割结果，错误发生的概率很小。因为我们的方法是面向对象的，我们的分割方法并不能够按照一个图片物体中相似组件的颜色图案进行分割。如图5.8显示了两张带有图案的示例图片。对于这两个柜子中的每一个，柜门都有着相似的形状，但颜色不同。同样的我们的系统无法对这种有着相似几何特征但颜色不同的组件进行分割。这也是我们方法的一个局限之处。



图 5.8: 橱柜上颜色图案的例子。



图 5.9: 在Google上使用关键词“chair”搜索到的图片。

使用本征图片对扩充我们的数据库很有帮助，因为提取出去掉光照的材质可以用于后续的渲染。然而这种方法并不够鲁棒，无法直接用于包含有大量图片的数据库中。在另一方面，本征图片分解对于我们的实验效果而言影响不大，因此我们没用在实验中使用这一方法。事实上我们后续建立的材质库也有着类

似的效果，我们也会进一步完善并使用该材质库中的材质进行渲染。

5.2.4 用户调查

我们同样发起了一项用户调查来研究用户是否能够将我们的场景着色结果与室内设计师所设计的结果区分出来。对于该用户调查，我们向每一位调查用户展示了一系列的渲染场景。这些图片是按照分组显示的，并且每一组分别包含了三组从相同的三维场景中提取的渲染结果，分别是我们的实验结果和两名受邀的室内设计师设计的结果。在本节的最后附有一些对比结果图。

用户调查细节。在用户调查中一共包含10组对比结果，5个场景。这些场景包括卧室，餐厅，客厅，会议室和办公室，并且都是在3ds Max中创建的。我们对每个场景都用一个特定的颜色主题作为输入，并使用我们的方法进行着色。我们手动地选择了5种颜色来生成视觉上美观的颜色主题，事实上已有多种方法可以用来生成和谐的颜色主题[15]，[37]，我们将在以后的工作中结合到我们的系统里。而室内设计师也按照同样的颜色主题为参照，选择其最中意的材质和颜色。对于每个场景使用我们方法得到的以及设计师设计的结果，都是在相同的视角和光照条件下，使用V-Ray3.3和3ds Max进行渲染出来的。

我们一共募集了83位调查对象。大多数都是计算机专业本科或研究生，都至少有一点图形学方面的背景基础。在这项用户调查中，每位对象都能看到这10组图片结果。我们确保的所有场景的出现顺序都是随机的，并且图片的摆放顺序也是随机的。调查对象需要为每张图片从1（糟糕）到5（极好的）进行打分，表示实验结果在视觉上的美观程度。

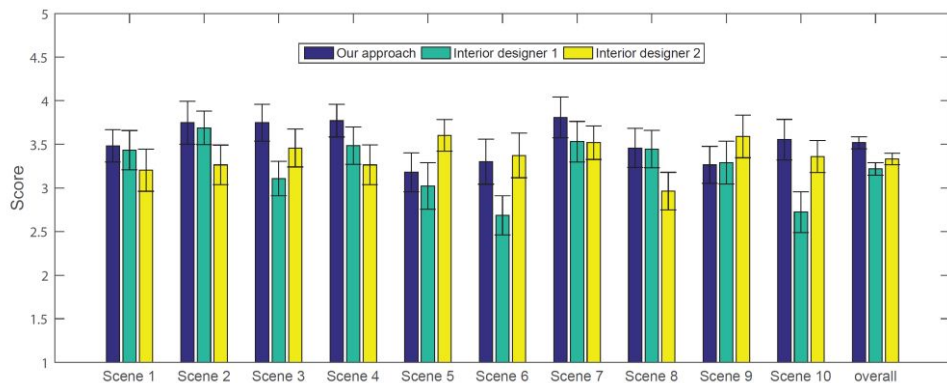


图 5.10: 对我们的结果和另外两位设计师的结果以95%置信区间统计得到的平均用户评分。

图5.10显示了以95%置信区间统计的每一组的平均分数。可以看出在7组实验结果中我们的结果稍微优于两位设计师的设计结果。我们进一步分析了统计

数据。对于每一组结果，如果调查对象对我们的实验结果的评分同时高于另外两组结果，则我们可以确信在该组结果中调查对象是更喜欢我们的结果。从总体上看，超过55.64%的调查对象相比于另外两组结果更倾向于选择我们的实验结果。但我们同邀请的设计师交流后他们表示，尽管我们的材质数据库中包含2650个材质采样，设计师仍觉得这些材质仍然不足，限制了他们在某些情况下的选择。他们希望能够提供类别更多的材质，并按照不同的设计风格进行分类。另外他们也认为选定了调色板也稍微限制了他们的选择。然而这项用户调查表明大多数的参与者无法将机器自动生成的着色结果和设计师设计出来的结果区分开来，也表明了我们实验方法的有效性。

Scene No.	Our approach	Designer 1	Designer 2
1	2.31s	27m48s	19m54s
2	3.38s	31m16s	16m57s
3	3.16s	23m37s	24m5s
4	3.32s	25m7s	23m32s
5	3.46s	27m57s	24m54s
6	3.1s	23m35s	22m17s
7	2.71s	26m24s	26m39s
8	3.18s	29m59s	27m10s
9	3.08s	24m6s	25m22s
10	2.78s	28m12s	24m29s

表 5.2: 我们的方法与室内设计师所花费的时间

如表5.2中所示，我们也同样统计了每位设计师处理输入的三维场景和为每个家具物体选择材质所花费的时间，不包括渲染所用的时间。平均下来每个场景大约需要30分钟的时间。我们在实验结束后与设计师们进行了交流，他们说尽管他们拥有丰富的经验，并且熟悉3ds Max，但小心地选择材质并将其赋予包含多个几何组件的材质模型仍然是一件令人痛苦并且费时的的工作。而与此形成对比的是，我们的方法大约每个场景只需要花费3秒的时间。

总而言之，用户调查表明我们的方法在室内设计方面可以发挥很好的作用。该方法可以将用户从费时费力的家具和场景着色任务中解脱出来。



图 5.11: 第一列图片为我们的实验结果, 其他两列分别为两位室内设计师设计的结果。

第六章 总结与展望

本文针对现有技术的不足，提供了一种数据驱动的室内场景着色方法，从而自动地为室内场景及其中的模型家具进行着色。其核心在于为场景中的每个家具求解最优的着色方案，并根据图片对家具网格模型进行分割，从而给各部分赋予相应的材质。

6.1 本文工作总结

我们介绍了一种新颖的方法，用来自动地为三维模型场景生成期望的颜色。我们的方法是面向对象的，即我们的方法能够根据家具图片为每个单独的家具进行着色。我们方法的核心是学习图片引导的模型分割。本文方法的流程是首先建立图像-模型数据库与纹理数据库；然后对图像-模型数据库中每一类家具的三维模型训练模型分类器；然后提取图像-模型数据库中家具的颜色主题并建立概率模型；接着根据建立好的概率模型以及用户输入的颜色主题求解出最优着色方案；然后是使用相应的分类器对输入场景中的每个3D模型进行分割；最后是根据得到的着色方案为输入场景中的每个家具赋予相应的材质。除了根据用户的喜好向用户推荐其想要的着色策略，本方法所具有的面向对象的特点也能够实现通过将一张示例图片的颜色转移到场景模型中进行着色。另外也允许用户通过交互式的方法，对场景中的任意物体的着色进行调整，而场景中的其他物体也会相应的适应该调整。我们的方法是基于数据驱动实现的。我们构建了一个层级式的图像-模型数据库，包含细致的标记信息，以及图片模型之间的对应关系，还包括一个材质数据库，可以贡献出来用于其他研究。我们的实验结果已经被证实与室内设计师的设计结果是相近的，人们甚至无法区分两者之间的不同。

6.2 未来工作展望

目前我们没有深入地去考虑关于室内设计和图形学设计一些更高层次的着色指导以及美学方面的规则,但因为我们所选取的网络中的专业图片本质上也是遵照这些规则的，基于数据驱动而实现的我们的方法因此也是严格遵守这些准则的，我们的实验结果以及用户调查验证了我们实验的有效性，当然后续我们

也可以通过一些指导规则来生成指定的颜色风格，比如中式或欧式。将来的后续工作可以显式地将这些标准整合进来，作为我们的目标方程中的额外项。另外我们将进一步完善我们所建立的材质库，将种类更多的材质添加进来，并使用这些材质结合法线信息，高光信息等将场景渲染得更加真实，进一步提升我们的室内场景着色效果。尽管我们已经开发出了一套流程来实现自动着色，对于初学者而言这仍是一个枯燥乏味的过程。场景理解，尤其是场景解析和物体识别，在最近几年取得了巨大进展[42]，[43]。在将来，我们也打算将这些内容整合到我们的流程中来协助用户进行交互。

参考文献

- [1] 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 Transactions on Graphics (TOG)*, vol. 30, no. 4, p. 86, 2011.
- [2] P. Merrell, E. Schkufza, Z. Li, M. Agrawala, and V. Koltun, “Interactive furniture layout using interior design guidelines,” *ACM Transactions on Graphics (TOG)*, vol. 30, no. 4, p. 87, 2011.
- [3] M. Fisher, D. Ritchie, M. Savva, T. Funkhouser, and P. Hanrahan, “Example-based synthesis of 3d object arrangements,” *ACM Transactions on Graphics (TOG)*, vol. 31, no. 6, p. 135, 2012.
- [4] S. Bell, P. Upchurch, N. Snavely, and K. Bala, “Opensurfaces: A richly annotated catalog of surface appearance,” *ACM Transactions on Graphics (TOG)*, vol. 32, no. 4, p. 111, 2013.
- [5] T. Welsh, M. Ashikhmin, and K. Mueller, “Transferring color to greyscale images,” *ACM Transactions on Graphics (TOG)*, vol. 21, no. 3, pp. 277 – 280, 2002.
- [6] R. Irony, D. Cohen-Or, and D. Lischinski, “Colorization by example,” in *Eurographics Symp. on Rendering*, vol. 2. Citeseer, 2005.
- [7] A. Levin, D. Lischinski, and Y. Weiss, “Colorization using optimization,” in *ACM Transactions on Graphics (TOG)*, vol. 23, no. 3. ACM, 2004, pp. 689 – 694.
- [8] Y. Qu, T.-T. Wong, and P.-A. Heng, “Manga colorization,” in *ACM Transactions on Graphics (TOG)*, vol. 25, no. 3. ACM, 2006, pp. 1214 – 1220.
- [9] Q. Luan, F. Wen, D. Cohen-Or, L. Liang, Y.-Q. Xu, and H.-Y. Shum, “Natural image colorization,” in *Proceedings of the 18th Eurographics conference on Rendering Techniques*. Eurographics Association, 2007, pp. 309 – 320.
- [10] D. Sykora, J. Burianek, and J. Zara, “Unsupervised colorization of black-and-white cartoons,” in *Proceedings of the 3rd international symposium on Non-photorealistic animation and rendering*. ACM, 2004, pp. 121 – 127.

- [11] A. Y.-S. Chia, S. Zhuo, R. K. Gupta, Y.-W. Tai, S.-Y. Cho, P. Tan, and S. Lin, “Semantic colorization with internet images,” in *ACM Transactions on Graphics (TOG)*, vol. 30, no. 6. ACM, 2011, p. 156.
- [12] S. Lin, D. Ritchie, M. Fisher, and P. Hanrahan, “Probabilistic color-by-numbers: Suggesting pattern colorizations using factor graphs,” *ACM Transactions on Graphics (TOG)*, vol. 32, no. 4, p. 37, 2013.
- [13] A. A. Gooch, S. C. Olsen, J. Tumblin, and B. Gooch, “Color2gray: salience-preserving color removal,” in *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3. ACM, 2005, pp. 634 – 639.
- [14] K. Rasche, R. Geist, and J. Westall, “Re-coloring images for gamuts of lower dimension,” in *Computer Graphics Forum*, vol. 24, no. 3. Wiley Online Library, 2005, pp. 423 – 432.
- [15] D. Cohen-Or, O. Sorkine, R. Gal, T. Leyvand, and Y.-Q. Xu, “Color harmonization,” in *ACM Transactions on Graphics (TOG)*, vol. 25, no. 3. ACM, 2006, pp. 624 – 630.
- [16] H.-Z. Huang, S.-H. Zhang, R. R. Martin, and S.-M. Hu, “Learning natural colors for image recoloring,” in *Computer Graphics Forum*, vol. 33, no. 7. Wiley Online Library, 2014, pp. 299 – 308.
- [17] Z. Yan, H. Zhang, B. Wang, S. Paris, and Y. Yu, “Automatic photo adjustment using deep neural networks,” *ACM Transactions on Graphics (TOG)*, 2015.
- [18] G. Leifman and A. Tal, “Mesh colorization,” in *Computer Graphics Forum*, vol. 31, no. 2pt2. Wiley Online Library, 2012, pp. 421 – 430.
- [19] A. Jain, T. Thormählen, T. Ritschel, and H.-P. Seidel, “Material memex: automatic material suggestions for 3d objects,” *ACM Transactions on Graphics (TOG)*, vol. 31, no. 6, p. 143, 2012.
- [20] C. H. Nguyen, T. Ritschel, K. Myszkowski, E. Eisemann, and H.-P. Seidel, “3d material style transfer,” in *Computer Graphics Forum*, vol. 31, no. 2pt2. Wiley Online Library, 2012, pp. 431 – 438.

- [21] K. Chen, K. Xu, Y. Yu, T.-Y. Wang, and S.-M. Hu, “Magic decorator: automatic material suggestion for indoor digital scenes,” *ACM Transactions on Graphics (TOG)*, vol. 34, no. 6, p. 232, 2015.
- [22] T. Y. Wang, H. Su, Q. Huang, J. Huang, L. Guibas, and N. J. Mitra, “Unsupervised texture transfer from images to model collections,” in *SIGGRAPH ASIA*, 2016.
- [23] L. Nan, K. Xie, and A. Sharf, “A search-classify approach for cluttered indoor scene understanding,” *ACM Transactions on Graphics (TOG)*, vol. 31, no. 6, p. 137, 2012.
- [24] Y. M. Kim, N. J. Mitra, D.-M. Yan, and L. Guibas, “Acquiring 3d indoor environments with variability and repetition,” *ACM Transactions on Graphics (TOG)*, vol. 31, no. 6, p. 138, 2012.
- [25] T. Shao, W. Xu, K. Zhou, J. Wang, D. Li, and B. Guo, “An interactive approach to semantic modeling of indoor scenes with an rgb-d camera,” *ACM Transactions on Graphics (TOG)*, vol. 31, no. 6, p. 136, 2012.
- [26] K. Chen, Y.-K. Lai, Y.-X. Wu, R. Martin, and S.-M. Hu, “Automatic semantic modeling of indoor scenes from low-quality rgb-d data using contextual information,” *ACM Transactions on Graphics (TOG)*, vol. 33, no. 6, p. 208, 2014.
- [27] P. Felzenszwalb and D. Huttenlocher, “Efficient graph-based image segmentation,” *International Journal of Computer Vision*, vol. 59, no. 2, pp. 167 – 181, 2004.
- [28] Y.-K. Lai, S.-M. Hu, R. R. Martin, and P. L. Rosin, “Fast mesh segmentation using random walks,” in *Proceedings of the 2008 ACM Symposium on Solid and Physical Modeling*, ser. SPM’ 08. New York, NY, USA: ACM, 2008, pp. 183 – 191.
- [29] L. Shapira, A. Shamir, and D. Cohen-Or, “Consistent mesh partitioning and skeletonisation using the shape diameter function,” *The Visual Computer*, vol. 24, no. 4, pp. 249 – 259, 2008.
- [30] A. Shamir, “A survey on mesh segmentation techniques,” in *Computer graphics forum*, vol. 27, no. 6. Wiley Online Library, 2008, pp. 1539 – 1556.

- [31] E. Kalogerakis, A. Hertzmann, and K. Singh, “Learning 3d mesh segmentation and labeling,” *ACM Transactions on Graphics (TOG)*, vol. 29, no. 4, p. 102, 2010.
- [32] M. Hilaga, Y. Shinagawa, T. Kohmura, and T. L. Kunii, “Topology matching for fully automatic similarity estimation of 3d shapes,” in *ACM Siggraph 2001*. New York, NY, USA: ACM, 2001, pp. 203 – 212.
- [33] S. Belongie, J. Malik, and J. Puzicha, “Shape matching and object recognition using shape contexts,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 509 – 522, Apr 2002.
- [34] A. Johnson and M. Hebert, “Using spin images for efficient object recognition in cluttered 3d scenes,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 5, pp. 433 – 449, May 1999.
- [35] A. Torralba, K. P. Murphy, and W. T. Freeman, “Sharing visual features for multiclass and multiview object detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 5, pp. 854 – 869, 2007.
- [36] B. Wang, Y. Yu, T.-T. Wong, C. Chen, and Y.-Q. Xu, “Data-driven image color theme enhancement,” in *ACM Transactions on Graphics (TOG)*, vol. 29, no. 6. ACM, 2010, p. 146.
- [37] P. O’ Donovan, A. Agarwala, and A. Hertzmann, “Color compatibility from large datasets,” in *ACM Transactions on Graphics (TOG)*, vol. 30, no. 4. ACM, 2011, p. 63.
- [38] W. K. Hastings, “Monte carlo sampling methods using markov chains and their applications,” *Biometrika*, vol. 57, no. 1, pp. 97 – 109, 1970.
- [39] J. S. Liu, *Monte Carlo strategies in scientific computing*. Springer Science & Business Media, 2008.
- [40] M. Schmidt, “Ugm: A matlab toolbox for probabilistic undirected graphical models,” <http://www.cs.ubc.ca/~schmidtm/Software/UGM.html>, 2010.
- [41] X. Chen, A. Golovinskiy, and T. Funkhouser, “A benchmark for 3d mesh segmentation,” *ACM Transactions on Graphics (TOG)*, vol. 28, no. 3, pp. 73:1 – 73:12, Jul. 2009.

- [42] L. Ladicky, C. Russell, P. Kohli, and P. Torr, “Associative hierarchical crfs for object class image segmentation,” in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 739 – 746.
- [43] M.-M. Cheng, S. Zheng, W.-Y. Lin, V. Vineet, P. Sturgess, N. Crook, N. J. Mitra, and P. Torr, “Imagespirit: Verbal guided image parsing,” *ACM Transactions on Graphics (TOG)*, vol. 34, no. 1, p. 3, 2014.
- [44] A. Quattoni, and A. Torralba. Recognizing Indoor Scenes. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

科研成果

攻读硕士学位期间完成的学术成果

1. Jie Zhu, Yanwen Guo, and Han Ma, "A Data-driven Approach for Furniture and Indoor Scene Colorization". Resubmitted to IEEE Transactions on Visualization and Computer Graphics.

发明专利

1. 马晗，郭延文，朱捷，夏元轶。发明专利：一种数据驱动的室内场景着色方法。中国发明专利（专利申请号：201610139954.X）。

致谢

首先感谢郭延文老师的悉心指导，郭老师在选题上提出了自己富有创造性的想法，为我确定了论文的核心内容，并在完成论文过程中一直给予指导；感谢朱捷师兄在核心算法实现上的帮助。论文中编程量很大，而且为了提升实验效果在算法的设计方面需要对图形学方面很深的理解，没有他的帮忙仅凭我一一人之力是无法完成的；感谢同级的吕高建同学，实现的材质扫描系统为我们的实验提供了材质库基础，并帮助我完善论文；感谢同级的贺敬武同学，在学习生活方面都给予我很多帮助。

在自动着色项目的实施中，感谢朱捷师兄，吕高建同学的大力帮助，使得项目能够顺利推进。同时感谢实验室的其余所有师兄师妹们，祝大家学业有成，工作顺利，梦想成真。也祝后来的师弟师妹们能拥有一段愉快而充实的研究生生活。感谢女友吕高建的陪伴与照顾，为我的研究生生活增添了许多色彩；感谢舍友蒋庆远和李鑫，以及所有其他一起本校考研的兄弟们，祝大家工作顺利。

最后要感谢家人的陪伴和照顾，给我力量与支持，让我一步一步地走到今天。