Room Layout Estimation with Object and Material Attributes Information using a Spherical Camera

Hansung Kim[1], Teofilo de Campos[1,2] and Adrian Hilton[1]
[1]Centre for Vision Speech and Signal Processing, University of Surrey, Guildford, UK
[2]FGA, University of Brasilia, Gama-DF, 72.444-240, Brazil
h.kim@surrey.ac.uk, t.decampos@st-annes.oxon.org, a.hilton@surrey.ac.uk

Abstract

In this paper we propose a pipeline for estimating 3D room layout with object and material attribute prediction using a spherical stereo image pair. We assume that the room and objects can be represented as cuboids aligned to the main axes of the room coordinate (Manhattan world). A spherical stereo alignment algorithm is proposed to align two spherical images to the global world coordinate system. Depth information of the scene is estimated by stereo matching between images. Cubic projection images of the spherical RGB and estimated depth are used for object and material attribute detection. A single Convolutional Neural Network is designed to assign object and attribute labels to geometrical elements built from the spherical image. Finally simplified room layout is reconstructed by cuboid fitting. The reconstructed cuboid-based model shows the structure of the scene with object information and material attributes.

1. Introduction

Estimating semantic room geometry is a classic problem in computer vision with a wide range of applications. There have been many studies into indoor scene geometry reconstruction from various sensors such as a photography camera, video camcorder and RGBD camera [7, 27, 4]. Recently this 3D geometry reconstruction evolved into semantic 3D scene reconstruction where the goal is not only to build geometry in 3D, but also to identify and localise known objects in the scene [26, 21]. Recognition of 3D objects and material is one of the classic problems using RGB [12, 1, 5] or RGB-D [10, 3] images. Survey of object classification in 3D range scans by Zelener [29] concludes that modelling contextual relations for structured prediction provides significant benefits to various applications.

However, current approaches using normal or RGBD cameras have the following limitations for complete indoor semantic scene reconstruction. First, indoor scenes generally include various sources of error in depth and geometry estimation. Textureless and non-Lambertian surfaces often result in errors in feature detection and matching. Highly reflective scenes with glass, mirrors or shiny surfaces can induce false depth. Second, normal or RGBD cameras have limited field-of-views (FOV) capturing only a part of the whole environment. For a complete scene layout estimation, multiple inputs and fusion technique are required.

In this paper, we propose a cuboid-based semantic room layout estimation pipeline using an off-the-shelf spherical 360° camera. This produces a complete scene model with semantic object and material attribute information. The approach assumes that room interiors are composed of piecewise planar surfaces aligned to the main axes (Manhattan world) as proposed in [6, 9]. Piecewise-planar scene elements are detected and aligned to the main axes using stereo matching, and their object classes and material attributes are predicted with a multi-scale Convolutional Neural Network (CNN). Finally simplified 3D scene structure with object and material labels is recovered by fitting cuboids into the reconstructed scene elements.

The main contributions of this paper are:

- A complete pipeline for approximate room geometry and object attribute estimation combining spherical stereo and CNN.
- A spherical stereo camera alignment algorithm for efficient and accurate depth estimation for off-the-shelf spherical cameras.
- Extension of the existing semantic labelling architecture with a multi-scale CNN for multi-class classification of object type and material attributes.

2. Related Work

2.1. Approximated room geometry reconstruction

Indoor 3D scene reconstruction has been a long-standing area of research. Multi-view stereo and structure from mo-

tion methods using multiple photos or videos have been widely investigated [25, 7]. As low-cost RGBD cameras have become readily available, various 3D reconstruction methods have been proposed using colour and range data. KinectFusion [20] made a great impact on real-time dense scene reconstruction with a RGBD camera and has been extended for large scale scene modelling. Public RGBD indoor datasets for the benchmark assessment have been also presented including ICL-NUIM [11], SUN3D [28], NYU [23, 24]. However, the limited FOV presents a challenging problem to ensure complete scene coverage for reconstruction as mentioned.

Spherical imaging provides a solution to overcome this coverage problem. Schoenbein et al. [22] proposed a high-quality omnidirectional 3D reconstruction of Manhattan worlds from catadioptric stereo video cameras. However, these catadioptric omnidirectional cameras have a large number of systematic parameters including the camera and mirror calibration. In order to get high resolution spherical images with simple and accurate calibration and matching, Point Grey developed an omnidirectional multi-camera system, the Ladybug[1]. Spheron developed a line-scan camera, Spheron VR[2], with a fish-eye lens to capture the full environment as an accurate high resolution / high dynamic range latitude-longitude image. Kim and Hilton used this Spheron VR for simplified scene modelling [15]. Li [17] has proposed a spherical image acquisition method using two video cameras with fish-eye lenses pointing in opposite directions. The biggest problem of the spherical stereo imaging from fish-eye lenses is large errors around epipoles and complex search along conic curves for stereo matching. This problem has been solved with accurate calibration and rectification. Various inexpensive off-the-shelf spherical cameras with two fish-eye lenses recently became popular in our daily lives[3][4][5]. Our room geometry modelling method used in this work is motivated from [15], but simplified for indoor room modelling and also extended to 3 DOF (roll, pitch and yaw) alignment for a commodity spherical camera.

2.2. Object and material attribute detection

Semantic segmentation methods aim to label every pixel in the image into a set of known classes. Zhu et al. [32] provide a good survey of semantic segmentation methods using RGB images. The use of RGBD images has a shorter history but a significant amount of works have already been presented [13, 10, 26, 3]. RGBD images carry more information but depth maps can be noisy and may contain large areas with missing measurements.

After the breakthrough results on ImageNet [16], the traditional pipeline of semantic object classification has been replaced by CNN [2]. CNNs are able to learn hierarchical representations that are customised for target applications. Recently CNNs have been used for semantic object detection and segmentation in various ways [31, 18, 8]. Eigen and Fergus [5] proposed an hierarchical fully convolutional networks (FCN) architecture composed of three scales. The first scale is VGG-FCN [18], and its output is up-sampled, concatenated with a higher resolution version of the input images at the next scale. The same process occurs at the interface between the second and third scales.

The problem of material attributes segmentation is similar to semantic object segmentation, except that each pixel can be assigned to multiple classes at the same time, e.g., the same surface can be wooden, hard, flat and be part of an object labelled as table. Zheng et al. [30] introduced the attributes NYU (aNYU) dataset which added 11 attribute labels to those of the NYU Depth v2 dataset of [24].

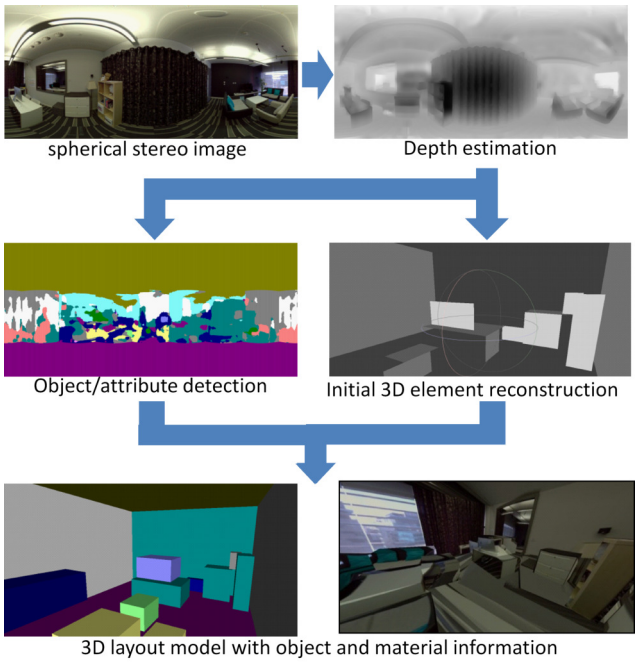


Figure 1. Block diagram of the proposed system

3. Proposed Method

3.1. Overview of the proposed pipeline

Figure 1 shows a block diagram for the whole process to build a structured room layout with object and material labels using cuboid scene/object proxies estimated from a

[1] Pointgrey, https://www.ptgrey.com/360-degree-spherical-camera-systems

[2] Spheron, https://www.spheron.com/products.html

[3] LG 360, http://www.lg.com/uk/lg-friends/lg-LGR105

[4] Samsung Gear 360, http://www.samsung.com/global/galaxy/gear-360/

[5] Ricoh Theta S, https://theta360.com/en/

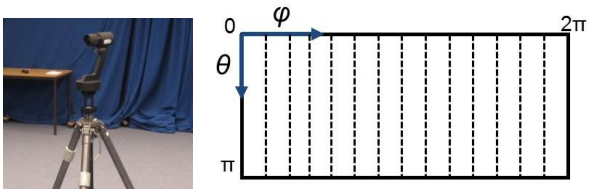
spherical stereo image pair. A full surrounding scene is captured by a spherical camera at two different heights as a vertical stereo pair. The captured spherical images are mapped to latitude-longitude (equirectangular) images and aligned to the room coordinate axes. Depth information of the scene is retrieved by stereo matching. Then the process is split into two: object/material detection and 3D element reconstruction. For input to the CNN in a standard perspective image format, the spherical image is projected onto a cube centred on the camera giving perspective images, with normal and depth. The predicted object and material labels from the CNN are back-projected to the original equirectangular format. In parallel, planar regions are detected from the spherical colour and depth information, and initial axis-aligned 3D planes are reconstructed. Finally object and material information are assigned to each 3D plane by voting, and cuboid proxies are fitted to the planes to generate a complete cuboid-based room layout model.

3.2. Spherical camera system and stereo alignment

Two different types of spherical cameras are used in this work. The first one is the Spheron VR, a mechanically tuned line-scan camera shown in Fig. 2 (a). The camera rotates on the axis passing through its optical centre, and a full spherical view is generated by mosaicing rays from its vertical slit. The fish-eye lens is pre-calibrated so that the rays through the vertical slit are evenly and accurately mapped from 0 to π on the image domain. Therefore the stitched image is an equirectangular projection image illustrated in Fig. 2 (b). However, Spheron VR is a high-end industrial camera which is expensive and takes a long time to scan a scene. The second type is the Theta S camera by Ricoh shown in Fig. 2 (c). Photos acquired from two pre-calibrated fish-eye lenses are stitched to each other to generate an equirectangular projection image as illustrated in Fig. 2 (d) (image from the Ricoh Theta SDK document). Projection from the Theta camera is less accurate than that from the Spheron VR, but it requires simple set up and captures a spherical photo or video in real-time.

To recover 3D information, the scene is captured with the spherical camera at two different heights. We use a vertical stereo system rather than typical horizontal stereo because depth error induced from stereo matching errors increases as the elevation angle to the baseline decreases as reported in [14]. This error diverges to the infinity on the epipoles (blind spot). The vertical stereo system makes these blind spots on the ceiling and floor which are less important and can be easily concealed by neighbouring information, while the horizontal stereo system makes the blind spots on the side which may include important scene information.

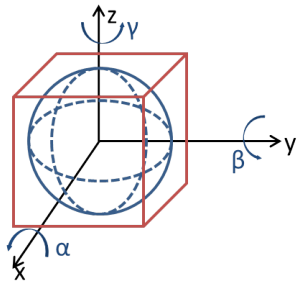
Even though the baseline of the vertical stereo camera system is perpendicularly aligned to the ground, the spherical coordinate of each spherical camera can be misaligned



(a) Spheron camera (b) Equirectangular projection image

(c) Theta camera (d) Projection of Theta camera

Figure 2. Camera system



(a) Spherical/cubic projection

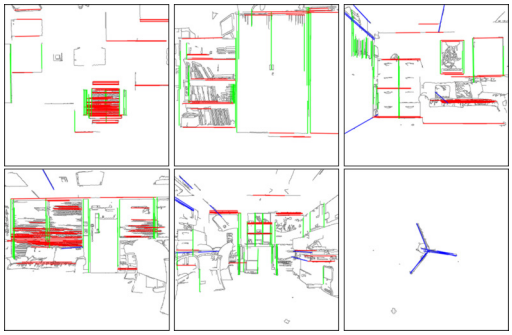
(b) Cubic projection images before alignment

(c) Cubic projection images after alignment

Figure 3. Spherical and Cubic projection

either to each other or to the world (room) coordinate system. For image alignment to the room coordinate, the equirectangular image in the Spherical coordinate is projected to a unit cube in the Cartesian domain fitted to the room coordinate as shown in Fig. 3 (a) and (b). If the spherical coordinate in aligned to the room coordinate, the horizontal and vertical lines in the scene are aligned to horizontal and vertical directions in each cubic projection image as shown in Fig. 3 (c). We utilise Hough line detection [19] in the cubic projection images to find the optimal rotation matrix for the coordinate alignment. The 3 DOF rotation

(a) Original stereo image pair (b) Aligned stereo image pair



(c) Hough lines detected in the cubic projection of the aligned image

Figure 4. Result of spherical image alignment

matrix can be obtained by the multiplication of single rotation matrices on x-axis (α), y-axis (β) and z-axis (γ) in Eq. (1), and the optimal α , β and γ values are found by Eq. (2), where k indexes the k-th face image in the cubic projection, H is lines detected by the Hough line detection, and C is cubic projection of the spherical image I. The Hough lines are categorised into general Hough line H, horizontal Hough lines H^h and vertical Hough lines H^v, where horizontal and vertical Hough lines represent detected Hough lines parallel and perpendicular to the horizon within 1° of angle difference.

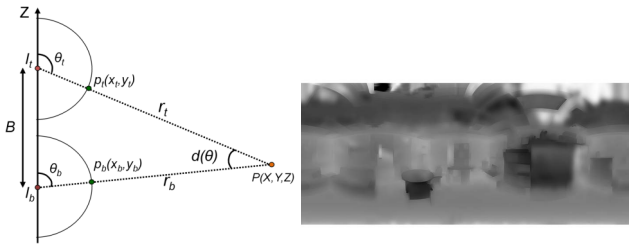
$$R(\alpha, \beta, \gamma) = R_x(\alpha)R_y(\beta)R_z(\gamma) \tag{1}$$

$$(\alpha_{opt}, \beta_{opt}, \gamma_{opt}) = \underset{\alpha,\beta,\gamma}{\operatorname{argmax}} \sum_{k=1}^{6} \frac{|H_k^h(\alpha, \beta, \gamma) \cup H_k^v(\alpha, \beta, \gamma)|}{|H_k(\alpha, \beta, \gamma)|} \tag{2}$$

$$H_k(\alpha, \beta, \gamma) = H(C_k(R(\alpha, \beta, \gamma)I(x, y, z)))$$

Finally, alignment between two vertical stereo images can be simply found by rotating one image by a multiple of 90° on the z-axis because both images have been aligned to the room coordinate.

Figure 4 shows an example of stereo alignment result. Red and green lines in Fig. 4 (c) represent H^h and H^v detected in the cubic projection images of the top image in Fig. 4 (b).



(a) Spherical stereo geometry (b) Disparity (depth) map

Figure 5. Depth reconstruction

3.3. Depth estimation and plane reconstruction

3D geometry of the scene is reconstructed using correspondence matching with spherical stereo geometry illustrated in Fig. 5 (a). Depth reconstruction from the aligned vertical spherical stereo images requires only baseline distance B and displacement of corresponding points. When disparity $d(\theta)$ as the angle difference between θ_b and θ_t, the distance of a certain 3D point P from the top camera is calculated as Eq. (3).

$$r_t = B / \left(\frac{\sin\theta_t}{\tan(\theta_t + d)} - \cos\theta_t \right) \tag{3}$$

Any correspondence matching algorithm can be used, but variational approaches are preferred rather than region-based matching algorithms because region-based methods suffer matching errors from spherical image distortion. We use a hierarchical PDE-based disparity estimation method [14] to produce smooth disparity fields with sharp depth discontinuities. Figure 5 (b) shows the disparity field from Fig. 4 (b). $0^\circ \leq \theta < 5^\circ$ and $175^\circ < \theta \leq 180^\circ$ regions have been cropped because depth from disparity diverge near the epipole areas (blind spots).

In order to build a piecewise planar elements in the scene from the estimated depth information, we utilise the block world reconstruction method proposed in [15]. One of the input spherical image is segmented into regions by the graph-based approach considering colour, surface normal and edge information, and optimised planes with fitted bounding boxes for each region are reconstructed. Reconstructed planes whose angles are not close to any of X-Y, Y-Z or X-Z planes are eliminated (violating Manhattan world assumption). Unreliable planes which are too distant from the camera or whose angle to the camera is too big are also eliminated. Close planes are merged into one plane to simplify the scene. Generated planes are back-projected to the original segmentation image to merge the segments for object and material attribute labelling.

3.4. Objects and material attributes detection

Our CNN architecture for semantic labelling was built on the design of [5]. It was modified for colour, depth and

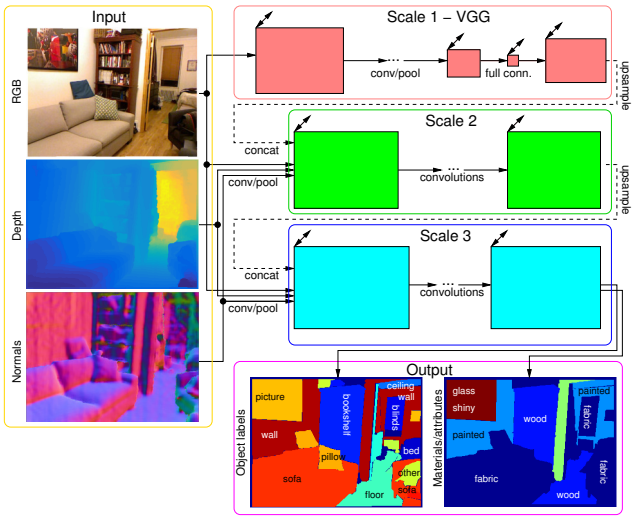


Figure 6. CNN architecture for multiple semantic labellings

surface normal inputs from stereo matching and adapted for multiple tasks: object and material attribute labelling. Figure 6 shows the modified CNN architecture.

Cubic projection images from the image alignment are used as the input of the CNN because the spherical image is not appropriate for this architecture due to its distortion from the spherical coordinate. Top and bottom images of the cubic projection have very little information for object recognition so they are forced to be labelled as "ceiling" and "floor", respectively.

In multiclass classification problems with neural networks, the loss function used for each prediction $\hat{y}$ is usually obtained using cross entropy:

$$\mathcal{L}(y, \hat{y}) = -y \cdot \log \hat{y} \,, \tag{4}$$

where ground truth labels $y \in \{0,1\}^C$ are binary vectors indicating the presence/abscence of each of the C classes and $\hat{y} \in [0,1]^C$ are class-based predictions, which are obtained by computing the softmax of the network's output.

For each batch of training samples, the losses are combined by:

$$\mathcal{L}(\mathtt{Y}, \hat{\mathtt{Y}}) = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(y^{(i)}, \hat{y}^{(i)}) \tag{5}$$

where N is the total number of pixels in the training batch.

In object detection and segmentation, each pixel is associated to a single class, i.e., $\sum_c^C \hat{y}_c = 1$. In material attribute detection, each pixel can be assigned to multiple labels, i.e., $\sum_c^C \hat{y}_c \in [0, C]$. Despite this difference, the same loss function of Eq. (4) can be used, but the expected value of loss of each task will be different, as that function depends on the number of classes and on the number times the ground truth $y = 1$ for each sample. Therefore, we propose to separately compute the loss Eq. (5) for each task t and combine them as follows:

$$\mathcal{L}(\mathtt{Y}, \hat{\mathtt{Y}}) = \sum_{t=1}^{T} \alpha_t \mathcal{L}(\mathtt{Y}^{(t)}, \hat{\mathtt{Y}}^{(t)}) \,, \tag{6}$$

where $\alpha_t \geq 0$ is the weight of each task such that $\sum_t^T \alpha_t = 1$, and $\mathtt{Y}^{(t)}$ are task-specific subparts of $\mathtt{Y}$ (the same goes for $\hat{\mathtt{Y}}^{(t)}$).

In other words, we assume that in our dataset, each sample is associated to labels of multiple tasks (objects and attributes) and that labels from all tasks are present for all training samples.

The CNN shares all parameters for all tasks up until the final layer, where task (and class) specific weights are present, as illustrated in Fig. 6.

In material attribute detection, instead of using the index of the maximum value of $\hat{y}_c$, a threshold τ is applied to the output of the classifier $\hat{y}$ and the resulting binary vector $\bar{y}_c$ is compared against y. To deal with the multiple labelling problem, we propose to explicitly learn a model of background pixels. Our classifier is trained with $C + 1$ class labels, where the first label is *none/background/unlabelled* and the remaining labels are those provided with the dataset. Therefore, instead of omitting unlabelled pixels from the loss function (Eq. (5)), we treat them as a new class and use their prediction value to set the attribute detection threshold, i.e., $\tau = \hat{y}_{\text{bgr}}$, and τ is set individually, for each pixel, rather than fixed to a predefined parameter. Any class whose probability is greater than that of the background is taken as detected in $\bar{y}_c$.

3.5. Final 3D room layout reconstruction

Objects and material attributes from the CNN architecture are used to vote to the corresponding regions of the back-projection of the reconstructed plane to decide the final labels for each plane. As a result, each plane has one object label and multiple attribute labels. Final 3D layout of the room is reconstructed by fitting cuboids into the plane elements as proposed in [15]. Objects and material labels are transferred to the cuboid elements.

In order to get a closed complete space of the room, the largest and farthest planes in each direction are considered as walls for the room layout and their surface normals are set to the inside of the room. All other planes are used for cuboid structure generation by the outward extrusion process from the camera capture position and the face normals are set outward of the cuboid.

(a) Usability room (b) Reception

Figure 7. Datasets

4. Experiments

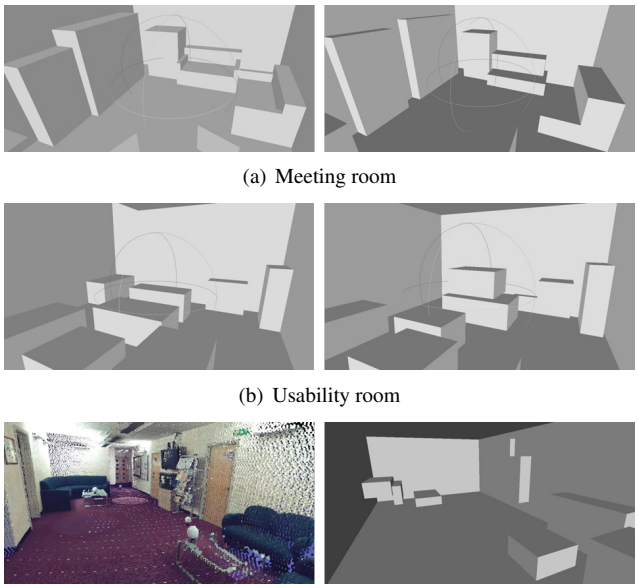
4.1. System set up and datasets

We tested two different spherical cameras introduced in Section 2: Spheron VR and Theta S. For the Spheron VR, we attached a 16mm fisheye lens and captured a vertical stereo pair with a baseline of 27cm. The resolution of spherical images is 3144×1414. The Theta S camera has its own built-in fisheye cameras which are internally calibrated. We captured the scene with the baseline distance of 11cm and resolution of 3000×1500.

We evaluated the proposed pipeline on three different indoor scenes: Meeting room (Fig. 4 (a), captured with Theta S), Usability room (Fig. 7 (a), captured with Spheron VR) and Reception (Fig. 7 (b), captured with Spheron VR). The Meeting room and Usability room are similar to normal living room environments in our daily lives, including various objects such as sofas, tables, bookcases, etc. The room sizes are 5.6m $\times$ 4.2m $\times$ 2.3m, and 5.6m $\times$ 5.2m $\times$ 2.9m, respectively. The Reception is not in a cuboid layout. The main area covers an area of 10.4m $\times$ 4.2m $\times$ 2.5m and it is connected to other corridors and rooms.

4.2. Room geometry modelling

Figure 8 (a) and (b) show the ground-truth models from the actual measurements and the reconstructed cuboid-based models from the spherical image pairs, respectively. The ground-truth models for Meeting room and Usability room were manually generated from the laser measurements, and the ground-truth model for Reception was acquired by a LIDAR scanner. The Meeting room data was captured by the Theta S camera which is less accurately rectified and aligned. Dimensions of the objects in the scene are slightly different from the ground-truth but the cuboid primitives represent the approximate structure of the scene well. The estimated room size is 6.15m $\times$ 4.7m $\times$ 2.5m which is slightly bigger than the ground-truth. In the result of the Usability room data, we can see that the room geometry is similar to the ground-truth. However, the thin monitor on the table which was neglected in the ground-truth model was reconstructed as a thick cuboid because the thickness could not be estimated from the images, and the table in the corner was missing because it was occluded by the monitor in the captured images. The estimated room size is 6.1m $\times$ 5m $\times$ 2.9m which is close to the true size. In the Reception dataset, furnitures and main area layout were well-estimated though opened doors and corridors to other rooms were missing. The estimated area size is 11.2m $\times$ 4.8m $\times$ 2.6m which is slightly bigger than the actual size.



(a) Meeting room

(b) Usability room

(c) Reception

Figure 8. Room geometry estimation results (Left: Ground-truth, Right: Reconstructed model

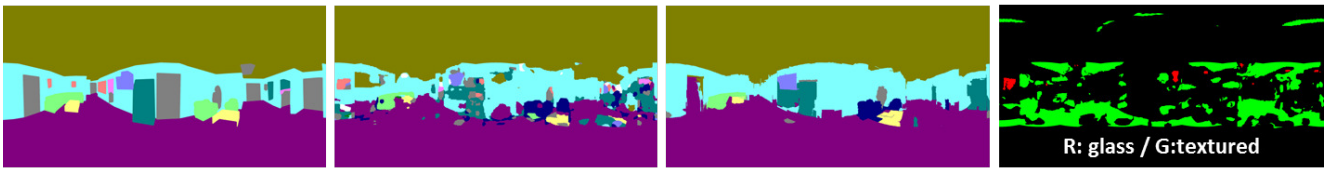
4.3. Object and material attribute labelling

In object labelling, we used the model of Eigen and Fergus [5] trained for the version of NYUDepth v2 dataset which was labelled with the 14 classes indexed in Fig. 9 (a). The training set consists of a set of 795 RGBD images, which was augmented using random transformations. The first to third columns of Fig. 9 (b)-(d) show manually annotated ground-truth, predicted labels from the CNN architecture and the final labels by voting to the reconstructed 3D plane elements. We can observe that cluttered labels due to lack of information or depth estimation error in the CNN outputs are refined to more semantic labels in the final results. To the best of our knowledge, this is the first work for semantic object labelling of spherical images, so it is difficult to compare its performance with other works. Figure 10 shows a 12 $\times$ 12 confusion matrix ("Bed" and "Unknown" labels were not considered). Most of the objects have been correctly classified but some false labels are observed in Sofa/Chair, Object/Furniture, Object/Wall, Picture/Wall and Wall/Furniture. In manual object annotation for the ground-truth generation, curtains and doors were annotated as "Object" because they are not in the original set of class labels. However, they are predicted as "Furniture" or "Wall" because they are located close to the wall. Pic-

(a) Object colour index

(b) Meeting room

(c) Usability room



(d) Reception

Figure 9. Object/material labelling results (First column: Object ground-truth, Second column: Object CNN output, Third column: Object final labels to 3D elements, Fourth column: Example of material detection)

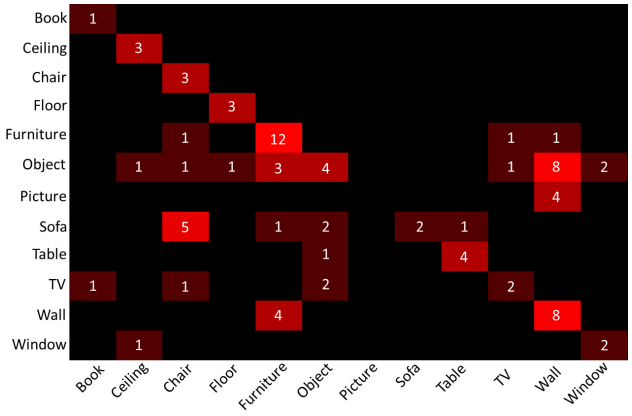


Figure 10. Confusion matrix (X:predicted, Y:Actual)

tures on the wall were also merged to "Wall" label in the final output due to the merge process in the 3D plane reconstruction.

In material attribute labelling, we initialised the CNN using the model that was pre-trained for object classification described above and fine-tunned it for the 11-class aNYU dataset[6] generated by Zheng et al. [30]. This was done using our multi-task loss function described in Sec. 3.4 (Eq. 6) iterating the learning process on the 724 RGBD samples of the aNYU training set for 500 epochs. Backpropagation and model updating was done in batches of 16 samples. This is a multi-label problem with material attribute labels of "'wood", "painted", "paper", "glass", "brick", "metal", "flat", "plastic", "textured", "glossy" and "shiny". We treat each attribute as a binary switch in a 12-bit vector which is "On" when its probability is higher than the probability of "none"[7]. It is hard to efficiently visualise multi-label images. The fourth column of Fig 9 shows examples of two selected material attributes represented in red and green channels. In the Meeting room set, many regions were labelled as "wood" and the frame on the wall and books in the bookcase were labelled as "paper". The bookcase region has

[6]aNYU dataset, `http://kylezheng.org/densesegattobj/`

[7]Pixels which did not have any label in the training set were labelled as "none" and treated as a standard class to be learnt, as discussed in Sec. 3.4.

	wood	painted	paper	glass	brick	metal	flat	plastic	textured	glossy	shiny
Wardrobe	V	V		V				V		V	V
Drawers	V	V					V	V		V	V
Picture	V		V	V							V
Sofa		V									
Door	V	V							V		
Table	V	V			V				V		
Chair	V	V							V		
Bookcase	V	V	V								

(a) Meeting room

	wood	painted	paper	glass	brick	metal	flat	plastic	textured	glossy	shiny
Wall		V			V				V		
Sofa	V	V									
Table	V	V									
Desk	V	V			V	V		V			
Drawers	V	V			V			V			
Bookcase	V	V								V	V
Curtains											

(b) Usability room

	wood	painted	paper	glass	brick	metal	flat	plastic	textured	glossy	shiny
Wall		V			V						
Sofa	V	V							V		
Table	V								V		
Bookstand											
Door	V	V							V	V	V
TV									V		
Plaque	V			V			V	V		V	V

(c) Reception

Figure 11. Predicted material attribute table for each object

both "wood" and "paper" labels. Lightings and some part of the floor were labels as "shiny" in the Usability room set, and the TV screen and plaque were labelled as "glass" in the Reception set. Figure 11 shows material attribute labels for the selected objects in the object ground-truth images. There are some mislabelling such as tables and desks with "brick", and failed material detection such as curtains and bookstand. However, most of the objects are labelled with reasonable attributes.

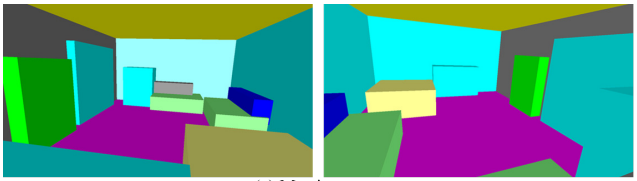
4.4. 3D layout with object and material information

For simple representation of the scene, all reconstructed cuboids with their object and material properties are saved as a vector list:

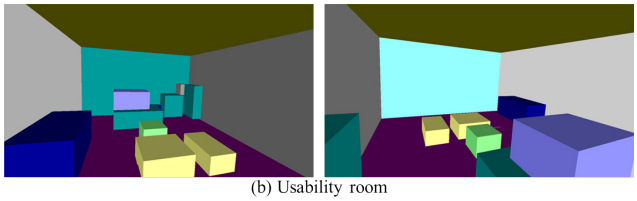
$$P = \{P_i\} = \{[T_i, B_x, B_y, B_z, O_i, M_i]\} \qquad (7)$$

where T_i is the type of element (invalid, plane and cuboid), $B_{x,y,z}$ are ranges to each direction, O_i is the object label and M_i is a 16 bit integer whose first 11 bits are used for binary material labels.

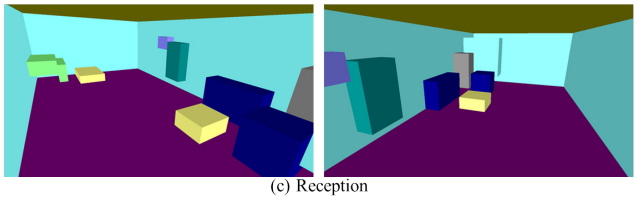
Figure 12 shows final room layouts with their labels from two different directions. The proposed method generated a coarse approximation of the scene structure with their object and material labels. A free-view rendering video of the scenes is available as supplemental material.



(a) Meeting room



(b) Usability room



(c) Reception

Figure 12. Final 3D room layout with object labels

5. Conclusions

In this work, we proposed a cuboid-based room layout and object/attribute estimation pipeline using a spherical camera. In the geometry estimation, a vertical spherical stereo capture generates texture with depth for the whole environment without any depth sensor. The captured images are aligned to the principal axes of the room coordinate and 3D plane elements are reconstructed. Semantic objects and material attributes in the scene are predicted by a CNN which was designed for multi-labelling problem with cubic projection images. The final cuboid-based room layout is reconstructed from the 3D planes labelled with object and material attribute. Results show that the proposed system generates compact representations of the room structures with object and material information. This work is still in progress and we believe this is a good step toward semantic 3D modelling with physical attributes.

Acknowledgements

This work was supported by the EPSRC Programme Grant S3A: Future Spatial Audio for an Immersive Listener Experience at Home (EP/L000539/1) and the BBC as part of the BBC Audio Research Partnership. Details about the data underlying this work are available from: http://dx.doi.org/10.17866/rd.salford.3822873. The authors would like to thank Sam Fowler for his contribution to the ground-truth data generation.

References

[1] M. Aubry, D. Maturana, A. Efros, B. Russell, and J. Sivic. Seeing 3d chairs: exemplar part-based 2d-3d alignment using a large dataset of cad models. In *Proc. CVPR*, 2014.

[2] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *Proc. BMVC*, 2014.

[3] K. Chen, Y.-K. Lai, and S.-M. Hu. 3D indoor scene modeling from RGB-D data: a survey. *Computational Visual Media*, pages 267–278, 2015.

[4] S. Choi, Q.-Y. Zhou, and V. Koltun. Robust reconstruction of indoor scenes. In *Proc. CVPR*, 2015.

[5] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proc. ICCV*, 2015.

[6] Y. Furukawa, B. Curless, S. Seitz, and R. Szeliski. Manhattan-world stereo. In *Proc. CVPR*, 2009.

[7] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski. Reconstructing building interiors from images. In *Proc. ICCV*, 2009.

[8] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik. Region-based convolutional networks for accurate object detection and segmentation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 38(1):142–158, 2016.

[9] A. Gupta, A. A. Efros, and M. Hebert. Blocks world revisited: Image understanding using qualitative geometry and mechanics. In *Proc. ECCV*, 2010.

[10] S. Gupta, P. Arbelez, R. Girshick, and J. Malik. Indoor scene understanding with rgb-d images: Bottom-up segmentation, object detection and semantic segmentation. *International Journal of Computer Vision*, pages 1–17, 2014.

[11] A. Handa, T. Whelan, J. McDonald, and A. Davison. A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM. In *Proc. ICRA*, 2014.

[12] Q. Hao, R. Cai, L. Zhang, Y. Pang, F. Wu, Y. Rui, and Z. Li. Efficient 2d-to-3d correspondence filtering for scalable 3d object recognition. In *Proc. CVPR*, 2013.

[13] O. Kahler and I. Reid. Efficient 3D scene labeling using fields of trees. In *Proc. iccv*, pages 3064–3071, 2013.

[14] H. Kim and A. Hilton. 3d scene reconstruction from multiple spherical stereo pairs. *International Journal of Computer Vision*, 104(1):94–116, 2013.

[15] H. Kim and A. Hilton. Block world reconstruction from spherical stereo image pairs. *Computer Vision and Image Understanding*, 139:104–121, 2015.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. NIPS*, pages 1097–1105, 2012.

[17] S. Li. Real-time spherical stereo. In *Proc. ICPR*, pages 1046–1049, 2006.

[18] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proc. CVPR*, 2015.

[19] J. Matas, C. Galambos, and J. Kittler. Robust detection of lines using the progressive probabilistic hough transform. *Computer Vision and Image Understanding*, pages 119–137, 2000.

[20] R. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Proc. ISMAR*, 2011.

[21] R. F. Salas-Moreno, R. A. Newcombe, H. Strasdat, P. H. J. Kelly, and A. J. Davison. Slam++: Simultaneous localisation and mapping at the level of objects. In *Proc. CVPR*, pages 1352–1359, 2013.

[22] M. Schoenbein and A. Geiger. Omnidirectional 3d reconstruction in augmented manhattan worlds. In *Proc. IROS*, pages 716 – 723, 2014.

[23] N. Silberman and R. Fergus. Indoor scene segmentation using a structured light sensor. In *Proc. ICCV Workshop*, 2011.

[24] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from RGBD images. In *Proc. ECCV*, pages 746–760, 2012.

[25] S. N. Sinha, D. Steedly, R. Szeliski, M. Agrawala, and M. Pollefeys. Interactive 3d architectural modeling from unordered photo collections. In *Proc. SIGGRAPH ASIA*, 2008.

[26] A. Wang, J. Lu, J. Cai, G. Wang, and T.-J. Cham. Unsupervised joint feature learning and encoding for rgb-d scene labeling. *IEEE Trans. Image Processing*, 24:4459–4473, 2015.

[27] J. Xiao and Y. Furukawa. Reconstructing the worlds museums. *International Journal of Computer Vision*, 110(3):243–258, 2014.

[28] J. Xiao, A. Owens, and A. Torralba. SUN3D: A database of big spaces reconstructed using sfm and object labels. In *Proc. ICCV*, pages 1625–1632, 2013.

[29] A. Zelener. Survey of object classification in 3d range scans. In *Technical report, City University of New York*, 2015.

[30] S. Zheng, M.-M. Cheng, J. Warrell, P. Sturgess, V. Vineet, C. Rother, and P. H. S. Torr. Dense semantic image segmentation with objects and attributes. In *Proc. CVPR*, 2014.

[31] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. Torr. Conditional random fields as recurrent neural networks. In *Proc. ICCV*, 2015.

[32] H. Zhu, F. Meng, J. Cai, and S. Lu. Beyond pixels: A comprehensive survey from bottom-up to semantic image segmentation and cosegmentation. *Journal of Visual Communication and Image Representation*, 34:12–27, 2016.