Unsupervised RGB-D image segmentation using joint clustering and region merging

Md. Abul Hasnat
mdabul.hasnat@univ-st-etienne.fr
Olivier Alata
olivier.alata@univ-st-etienne.fr
Alain Trémeau
alain.tremeau@univ-st-etienne.fr

Hubert Curien Lab., UMR CNRS 5516, Jean Monnet University, Saint Etienne, France.

Abstract

Recent advances in imaging sensors, such as Kinect, provide access to the synchronized depth with color, called RGB-D image. In this paper, we propose an unsupervised method for indoor RGB-D image segmentation and analysis. We consider a statistical image generation model based on the color and geometry of the scene. Our method consists of a joint color-spatial-axial clustering method followed by a statistical planar region merging method. We evaluate our method on the NYU depth database and compare it with existing unsupervised RGB-D segmentation methods. Results show that, it is comparable with the state of the art methods and it needs less computation time. Moreover, it opens interesting perspectives to fuse color and geometry in an unsupervised manner.

1 Introduction

Image segmentation is one of the most widely studied problems that groups perceptually similar pixels based on certain features (e.g. color, texture etc.) [30]. Numerous researches [8, 12, 15, 25, 31] have shown that the use of depth as an additional feature improves accuracy of scene segmentation. However, it remains an important issue - what is the best way to fuse color and geometry in an unsupervised manner? We focus on this issue and propose a solution.

A common approach for RGB-D segmentation is to extract different features, design kernels and classify pixels with learned classifiers. Ren *et al*. [25] proposed contextual models that combine kernel descriptors with a segmentation tree or with superpixels. For this task, they extended the well-known gPb-UCM algorithm [2] for RGB-D image. The method of Silberman *et al*. [27] starts from superpixels, aligns them with 3D planes, and finally applies a hierarchical segmentation using a trained classifier. Gupta *et al*. [12] first compute gPb [2] from a combination of geometric and monocular contour cues, then detect contours via a learned classifier and finally generate a hierarchy of segmentation. These methods are supervised i.e. require training from ground truth.

Among the unsupervised methods, Dal Mutto *et al*. [8] fuse color with 3D position using a multiplier and then apply Normalized Cut (N-Cut) method to cluster pixels. Taylor *et al*. [31] first extract edges, construct a triangular graph and apply N-Cut on the graph. Next,

they extract planar surfaces from the segments using RANSAC [30] and finally merge the co-planar segments using a greedy merging. Beside these, several methods [22, 29] extend the graph based segmentation [9] in order to fuse color with depth.

In this paper, we propose an unsupervised (i.e. no training) scene segmentation method that combines a clustering method with a region merging method. Our method first identifies the possible image regions using clustering w.r.t. a statistical image generation model and then merges regions based on planar statistics. The image model is based on three features[1]: color, 3D position and surface normal. It assumes that these features are issued independently (*naïve Bayes* [19] assumption) from a finite mixture of probability distributions.

Finite Mixture Models are often used for cluster analysis [4, 6, 19]. In image analysis and segmentation these models have been employed with the Gaussian distribution to cluster the image pixels [1, 11, 16, 20]. Our image model considers the Gaussian [19] distribution for color and 3D position and the Watson Distribution (WD) [28] for surface normal. We use WD because it overcomes the directional ambiguity and noise [13, 26] related to surface normal. Moreover, it provides adequate statistics to explain the planar geometry of regions, see [14] for more details.

We exploit *Bregman Soft Clustering (BSC)* [4] to cluster pixels w.r.t. our image model. BSC is a centroid based parametric clustering method which has been effectively employed for mixture models based on exponential family of distributions [21]. Compared to the traditional Expectation Maximization algorithms, BSC provides additional benefits: (a) it considers *Bregman Divergence* that generalizes a large number of distortion functions [4]; (b) simplifies computationally expensive Maximization step and (c) is applicable to mixed data type.

Existing region merging methods [18, 23, 24, 32] exploit color and edge. For indoor scenes, the use of color is often unreliable due to numerous effects caused by spatially varying illumination [12]. On the other hand, the planar surfaces are important geometric primitives which are often employed for scene decomposition [12, 26, 27] and grouping [31]. This motivates us to develop a region merging method based on planar property rather than color.

We can summarize our contributions as follows: (a) we propose a statistical RGB-D image generation model (Sec. 2.1) that incorporates both color and geometry of a scene; (b) we develop an efficient soft clustering method (Sec. 2.2) by exploiting the Bregman Divergence [4] to cluster heterogeneous data w.r.t. the image model; (c) we propose a statistical region merging method (Sec. 2.3) based on planar geometry, which can be used with other RGB-D segmentation methods and (d) we provide a benchmark (Sec. 3) on the NYU depth database V2 (NYUD2) [27] using standard evaluation metrics [2, 10]. Results show that our method is comparable with the state of the art and better w.r.t. computation time.

In the rest of the paper we describe our proposed method in Section 2, present the experimental results with discussion in Section 3 and finally draw conclusions in Section 4.

[1]Clustering using only 3D points often fails to locate the intersections among the planar surfaces with different orientations such as wall, floor, ceiling, etc. This is due to the fact that the 3D points associated to the intersections are grouped into a single cluster. On the other hand, the use of only normals groups multiple objects with nearly similar orientations into the same cluster irrespective of their 3D location. In order to overcome these limitations and to describe the geometry of indoor scenes, we take both features into account.

2 Methodology

2.1 Image Generation Model and Segmentation Method

We propose a statistical image model that fuses color and shape (3D and surface normal) features. The model assumes that the features are independently issued from a finite mixture of multivariate Gaussian (for color and 3D) and a multivariate Watson distribution (for surface normal). Mathematically, such a model with k components has the following form:

$$g(\mathbf{x}_i|\Theta_k) = \sum_{j=1}^{k} \pi_{j,k} f_g(\mathbf{x}_i^C|\mu_{j,k}^C, \Sigma_{j,k}^C) f_g(\mathbf{x}_i^P|\mu_{j,k}^P, \Sigma_{j,k}^P) f_w\left(\mathbf{x}_i^N|\mu_{j,k}^N, \kappa_{j,k}^N\right) \quad (1)$$

Here $\mathbf{x}_i = \{\mathbf{x}_i^C, \mathbf{x}_i^P, \mathbf{x}_i^N\}$ is the feature vector of the ith pixel with $i = 1, ..., M$. Superscripts denote: C - color, P - 3D position and N - normal. $\Theta_k = \{\pi_{j,k}, \mu_{j,k}^C, \Sigma_{j,k}^C, \mu_{j,k}^P, \Sigma_{j,k}^P, \mu_{j,k}^N, \kappa_{j,k}^N\}_{j=1...k}$ denotes the set of model parameters where $\pi_{j,k}$ is the prior probability, $\mu_{j,k}$ is the mean, $\Sigma_{j,k}$ is the variance-covariance matrix and $\kappa_{j,k}$ is the concentration of the jth component. $f_g(.)$ and $f_w(.)$ are the density functions of the multivariate Gaussian distribution (Section 2.2.2) and the multivariate Watson distribution (Section 2.2.3) respectively.

Fig. 1 illustrates the work flow of our RGB-D segmentation method that consists of two tasks: (1) cluster features and (2) merge regions. The first task performs a joint color-spatial-axial clustering and generates a set of regions. The second task performs a refinement on the set with the aim to merge regions which are susceptible to be over-segmented. In the next two sub-sections we present our methods to accomplish these tasks.

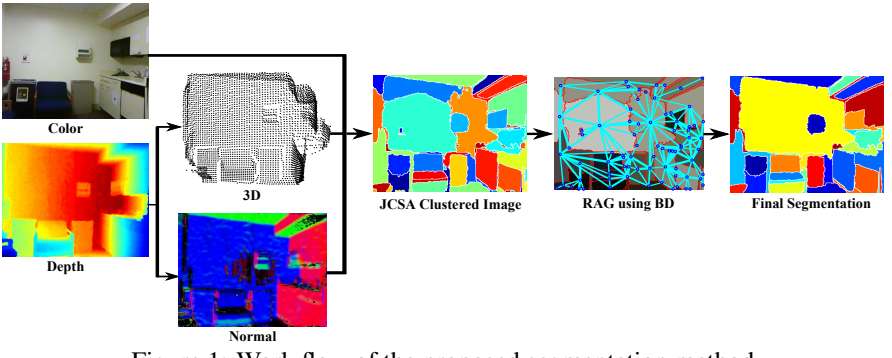

Figure 1: Work flow of the proposed segmentation method.

2.2 Joint Color-Spatial-Axial (JCSA) clustering

We develop a Joint Color-Spatial-Axial (JCSA) clustering method that estimates the parameters of the mixture model (Eq. (1)), clusters the pixels and hence provides the regions in the image. However, notice that in an unsupervised setting the true number of segments are unknown. Therefore, we assume a certain maximum number of clusters ($k = k_{max}$). Such an assumption often causes an over-segmentation of the image. In order to tackle this issue, it is necessary to merge the over-segmented regions (see Sec. 2.3).

2.2.1 Exponential Family of Distributions (EFD) and Bregman Divergence

A multivariate probability density function $f(x|\eta)$ belongs to the exponential family if it has the following (Eq. (3.7) of [4], Eq. (60) of [21]) form[2]:

$$f(x|\eta) = \exp(-D_G(t(x),\eta))\exp(k(x)) \tag{2}$$

and

$$D_G(\eta_1,\eta_2) = G(\eta_1) - G(\eta_2) - \langle \eta_1 - \eta_2, \nabla G(\eta_2)\rangle \tag{3}$$

with $G(.)$ the Legendre dual of log normalizing function which is a strictly convex function. ∇G the gradient of G. $t(x)$ denotes the sufficient statistics and $k(x)$ is the carrier measure. The expectation of the sufficient statistics $t(x)$ w.r.t. the density function (Eq. (2)) is called the expectation parameter (η). D_G is the Bregman divergence computed from expectation parameters: it can be used to compute the distance between two distributions of the same exponential family, defined by two expectation parameters η_1 and η_2. We give now the particular forms obtained with the Gaussian distribution and the Watson distribution.

2.2.2 Multivariate Gaussian Distribution

For a d dimensional random vector $\mathbf{x} = [x_1,...,x_d]^T \in \mathbb{R}^d$, the multivariate Gaussian distribution is defined as [21]:

$$f_g(\mathbf{x}|\mu,\Sigma) = \frac{1}{(2\pi)^{d/2}\det(\Sigma)^{1/2}}\exp\left(-\frac{1}{2}(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu)\right) \tag{4}$$

Here, $\mu \in \mathbb{R}^d$ denotes the mean and Σ denotes the variance-covariance symmetric positive-definite matrix. To write the multivariate Gaussian distribution in the form of Eq. (2), the elements are defined as [21]: sufficient statistics $t(\mathbf{x}) = (\mathbf{x}, -\mathbf{x}\mathbf{x}^T)$; carrier measure $k(\mathbf{x}) = 0$; expectation parameter $\eta = (\phi,\Phi) = \left(\mu, -(\Sigma + \mu\mu^T)\right)$ and $G_g(\eta) = -\frac{1}{2}\log(1+\phi^T\Phi^{-1}\phi) - \frac{1}{2}\log(\det(\Phi)) - \frac{d}{2}\log(2\pi e)$.

2.2.3 Multivariate Watson Distribution

For a d dimensional unit vector $\mathbf{x} = [x_1,...,x_d]^T \in S^{d-1} \subset \mathbb{R}^d$ (i.e. $\|\mathbf{x}\|_2 = 1$), the multivariate (axially symmetric) Watson distribution (mWD) is defined as [17]:

$$f_w(\mathbf{x}|\mu,\kappa) = M(1/2,d/2,\kappa)^{-1}\exp\left(\kappa(\mu^T\mathbf{x})^2\right) = f_w(-\mathbf{x}|\mu,\kappa) \tag{5}$$

Here, μ is the mean direction (with $\|\mu\|_2 = 1$), $\kappa \in \mathbb{R}$ the concentration and $M(1/2,d/2,\kappa)$ the Kummer's function [17, 28]. To write the mWD in the form of Eq. (2), the elements are defined as: sufficient statistics $t(\mathbf{x}) = \left[x_1^2,...,x_d^2,\sqrt{2}x_1x_2,...,\sqrt{2}x_{d-1}x_d\right]^T$; carrier measure $k(\mathbf{x}) = 0$; expectation parameter η as:

$$\eta = \|\eta\|_2 \nu \tag{6}$$

[2] In order to keep our formulations concise, we use the expectation parameters η to define the Exponential Family of Distributions. However, the other form [4, 10, 19, 21] $f(x|\theta) = \exp(\langle t(x),\theta\rangle - F(\theta) + k(x))$ and related derivations are available in [14] or in the supplementary materials.

where $\nu = \left[\mu_1^2, ..., \mu_d^2, \sqrt{2}\mu_1\mu_2, ..., \sqrt{2}\mu_{d-1}\mu_d\right]^T$ and

$$G_w(\eta) = \kappa \|\eta\|_2 - \log M(1/2, d/2, \kappa) \tag{7}$$

With the above formulation, for a set of observations $\chi = \{\mathbf{x}_i\}_{i=1,...,M}$ we estimate $\eta = E[t(\chi)]$ and κ with a Newton-Raphson root finder method as [28]:

$$\kappa_{l+1} = \kappa_l - \frac{g(1/2, d/2; \kappa_l) - \|\eta\|_2}{g^{'}(1/2, d/2; \kappa_l)} \tag{8}$$

where $g(1/2, d/2; .)$ is the Kummer-ratio, $g^{'}(1/2, d/2; .)$ is the derivative of $g(1/2, d/2; .)$. See [14] or the supplementary materials for additional details.

2.2.4 Bregman Divergence for the combined model

Our image model (in Eq. (1)) combines different exponential family of distributions (associated to color, 3D and normal) based on independent (*naïve Bayes* [19]) assumption. Therefore, Bregman Divergence (BD) of the combined model can be defined as a linear combination of the BD of each individual distributions:

$$D_G^{comb}(\eta_i, \eta_j) = D_{G,g}^C(\eta_i^C, \eta_j^C) + D_{G,g}^P(\eta_i^P, \eta_j^P) + D_{G,w}^N(\eta_i^N, \eta_j^N) \tag{9}$$

where, $D_{G,g}(.,.)$ denotes BD using multivariate Gaussian distribution and $D_{G,w}(.,.)$ denotes BD using multivariate Watson distribution. Then, it is possible to define, with expectation parameter $\eta = \{\eta^C, \eta^P, \eta^N\}$:

$$G^{comb}(\eta) = G_g(\eta^C) + G_g(\eta^P) + G_w(\eta^N) \tag{10}$$

2.2.5 Bregman Soft Clustering for the combined model

Bregman Soft Clustering exploits Bregman Divergence (BD) in the Expectation Maximization (EM) framework [19] to compute the Maximum Likelihood Estimate (MLE) of the mixture model parameters and provides a soft clustering of the observations [4]. In the expectation step (E-step) of the algorithm, the posterior probability is computed as [21]:

$$p(\gamma_i = j|\mathbf{x}_i) = \frac{\pi_{j,k} \exp\left(G^{comb}(\eta_{j,k}) + \left\langle t(\mathbf{x}_i) - \eta_{j,k}, \nabla G^{comb}(\eta_{j,k})\right\rangle\right)}{\sum_{l=1}^{k} \pi_{l,k} \exp\left(G^{comb}(\eta_{l,k}) + \left\langle t(\mathbf{x}_i) - \eta_{l,k}, \nabla G^{comb}(\eta_{l,k})\right\rangle\right)}, \; j = 1, ..., k \tag{11}$$

Here, $\eta_{j,k}$ and $\eta_{l,k}$ denote the expectation parameters for any cluster j and l given that the total number of components is k. The maximization step (M-step) updates the mixing proportion and expectation parameter for each class as:

$$\pi_{j,k} = \frac{1}{M}\sum_{i=1}^{M} p(\gamma_i = j|\mathbf{x}_i) \text{ and } \eta_{j,k} = \frac{\sum_{i=1}^{M} p(\gamma_i = j|\mathbf{x}_i)\mathbf{x}_i}{\sum_{i=1}^{M} p(\gamma_i = j|\mathbf{x}_i)} \tag{12}$$

Initialization is a prominent issue and has significant impact on clustering. Our initialization procedure consists of setting initial values for prior class probability $(\pi_{j,k})$ and the expectation parameters $(\eta_{j,k})$ with $1 \leq j \leq k$. We initialize π and η associated to the Gaussian and Watson distributions using a combined k-means type clustering. After initialization,

we iteratively apply the E-step and M-step until the convergence criteria are met. These criteria are based on maximum number of iterations (e.g. 200) and a threshold difference (e.g. 0.001) between the negative log likelihood values (see Eq. (13)) of two consecutive steps.

$$nLLH = -\sum_{i=1}^{M} \log\left(g\left(\mathbf{x}_i|\Theta\right)\right) \tag{13}$$

The above procedures lead to a soft clustering algorithm, which generates associated probability and parameters for each components of the proposed model in Eq. (1). Finally, for each sample we get the class label ($\hat{\gamma}_i$) using the updated combined BD (9):

$$\hat{\gamma}_i = \underset{j=1,\ldots,k}{\arg\min} D_G^{comb}(t(x_i), \eta_{j,k}) \tag{14}$$

2.3 Region Merging

In the previous step we cluster pixels with a high number of components, which causes over-segmentation. Therefore, we need to merge the over-segmented regions. To this aim, first we build a Region Adjacency Graph (RAG) [32] (see Fig. 1) by considering that each region is a node and each node has edges with its adjacent nodes. Then, similar to the standard region merging methods [23, 24, 32], we define a region merging predicate and merging order.

2.3.1 Region Adjacency Graph (RAG)

Let $R = \{r_i\}_{i=1,\ldots,M}$ be the set of regions that we obtain from the JCSA clustering; $G = (V,E)$ be an undirected graph represents the RAG, where $V = \{v_i\}_{i=1,\ldots,M}$ is the set of nodes corresponding to R and E is the set of edges among adjacent nodes. Each node v_i consists of the parameters (μ and κ) of the Watson distribution (Sec. 2.2.3) associated with region r_i. Each edge e_{ij} consists of two weights: w_d, based on statistical dissimilarity and w_b, based on boundary strength between adjacent nodes v_i and v_j. The weight w_d is defined as:

$$w_d(v_i, v_j) = \min(D_{G,w}^N(\eta_i^N, \eta_j^N), D_{G,w}^N(\eta_j^N, \eta_i^N)) \tag{15}$$

where, $D_{G,w}^N(\eta_i^N, \eta_j^N)$ is the Bregman Divergence (Eq. (3)) among the Watson distributions associated with regions r_i and r_j. The weight w_b is defined as:

$$w_b(v_i, v_j) = \frac{1}{|r_i \bigcap r_j|} \sum_{b \in r_i \bigcap r_j} I_G^{rgbd}(b) \tag{16}$$

where, $r_i \bigcap r_j$ is the set of boundary pixels among two regions, $|.|$ denotes the cardinality and I_G^{rgbd} is the normalized magnitude of image gradient (MoG) [30] computed from the RGB-D image. I_G^{rgbd} is obtained by first computing MoG for each color channels (I_G^r, I_G^g, I_G^b) and depth (I_G^d) individually, and then taking the maximum of those MoGs at each pixel.

2.3.2 Merging Strategy

Our merging strategy is an iterative procedure that proceeds by employing merging predicate among adjacent nodes in a certain order. Once two nodes are merged, the information

regarding the merged node and its edges are updated immediately. This procedure continues until no valid candidates are left to merge. We define the ***region merging predicate*** P_{ij} as:

$$P_{ij} = \begin{cases} true, & \text{if } (a)\ \kappa_j > \kappa_p \text{ and} \\ & (b)\ w_d(v_i,v_j) < th_d \text{ and } w_b(v_i,v_j) < th_b \text{ and} \\ & (c)\ planar\,outlier\,ratio > th_r; \\ false, & \text{otherwise.} \end{cases} \tag{17}$$

where κ_j is the concentration (sec. 2.2.3) of region r_j. κ_p is the threshold to define the planar property of a region, th_d and th_b are the thresholds associated with the distance weight w_d (Eq. (15)) and boundary weight w_b (Eq. (16)). th_r is the threshold associated with the plane outlier ratio, which is computed by first fitting a plane to the 3D points with RANSAC and then compute the ratio of inliers and outliers [31]. See Sec. 3 for details of these thresholds.

The predicate in Eq. (17) evaluates the *candidacy* (condition - (a)) of each node, *eligibility* (condition - (b)) of merging a pair of nodes and *consistency* (condition - (c)) of the merged node. ***candidacy*** of a node defines if it belongs to a planar surface. To this aim, we analyze the concentration (κ) associated to each node. This helps us to simplify the RAG and filter out a number of nodes and hence reduce the computational time. ***eligibility*** of a pair of nodes determines whether they should be merged. We exploit the edge weights (w_d and w_b) of the RAG in order to check this condition. ***consistency*** is applied to a merged region in order to check whether it remains a planar surface.

The ***region merging order*** [24] sorts the adjacent regions that should be evaluated and merged sequentially. However, it changes dynamically after each merging occurs. We define the *merging order* based on dissimilarity based weights w_d (Eq. 15) among the adjacent nodes. The adjacent node v_j which has minimum $w_d(v_i,v_j)$ is considered to be evaluated first. We use w_d as the merging order constraint due to its ability to provide a measure of dissimilarity among regions. Such a measure is based on the mean direction (μ) and the concentration (κ) of the surface normals of the regions. Therefore, with this constraint, the neighboring region which is most similar w.r.t. μ and κ will be selected as the first candidate to evaluate using Eq. (17).

3 Experiments and Results

In this section, we evaluate the proposed method on the benchmark image database NYUD2 [27] which consists of 1449 indoor images with RGB, depth and ground-truth information. We convert (using MATLAB function) the RGB color information into $L^*a^*b^*$ (CIELAB space) color because of its perceptual accuracy [2]. From the depth images, we compute the 3D coordinates and surface normals using the toolbox available with the database [27].

Our clustering method requires to set initial labels of the pixels and the number of clusters k. We initialize it following the k-means++ [3] strategy with $k = 20$. For the region merging we empirically set the thresholds as: $\kappa_p = 5$ to decide a region as planar, $th_b = 0.2$ to decide the existence of boundary among two regions, $th_d = 3$ to decide the distance among two regions and $th_r = 0.9$ to determine the goodness of a plane fitting.

We evaluate performance using the standard benchmarks [2] which are applied between the test and ground truth segmentation: (1) Probability Rand Index (*PRI*), it measures likelihood of a pair of pixels that has same label; (2) Variation of Information (*VoI*), it measures the distance between two segmentations in terms of their average conditional entropy; (3)

Boundary Displacement Error (*BDE*) [10], it measures the average displacement between the boundaries of two segmentations; (4) Ground Truth Region Covering (*GTRC*), it measures the region overlaps between ground truth and test and (5) Boundary based F-Measure (*BFM*), a boundary related measure based on precision-recall framework [2]. With these criteria a segmentation is better if *PRI*, *GTRC*, *BFM* are larger and *VoI* and *BDE* are smaller.

First we study the sensitivity of the proposed method w.r.t. the parameters (k, κ_p, th_b, th_d), which is presented in table 1. The parameter k belongs to the clustering (sec 2.2) while κ_p, th_b and th_d belong to the region merging method (sec 2.3). Note that, the parameter $th_r = 0.9$ is set by following [31] and hence we do not analyze it further. From table 1, we observe that while PRI (1%) is quite stable, VoI (6%), BDE (8%) and GTRC (7%) provide discriminating view w.r.t the parameters. The parameter k is inversely related to the number of pixels in a cluster. In segmentation, a smaller k causes to loose details in the scene while higher k splits the scene into more regions. We set κ_p based on a study on NYUD2 (see supplementary materials for details) which reveals that planar surfaces can be characterized with concentration $\kappa >= 5$. While, a lower κ value selects non-planar surfaces to be merged, a higher value may reject true planar surfaces for merging. Following the OWT-UCM [2] method, we empirically set the value of th_b. Similarly, we set th_d empirically. In theory two regions which belong to the same direction have a negligible value of Bregman divergence. However, the inaccurate computation of the shape features and the presence of noise in the acquired depth information often cause this divergence measure to be high. From our experience with the images of NYUD2, th_d should be within the range between 2 to 4.

	$\{k,5,0.2,3\}$			$\{20,\kappa_p,0.2,3\}$			$\{20,5,th_b,3\}$			$\{20,5,0.2,th_d\}$		
	15	20	25	2	5	8	0.1	0.2	0.3	2	3	4
PRI	0.89	0.90	0.89	0.89	0.90	0.90	0.89	0.90	0.89	0.90	0.90	0.90
VoI	2.31	2.29	2.42	2.32	2.29	2.38	2.43	2.29	2.32	2.37	2.29	2.32
BDE	10.64	9.83	10.05	10.52	9.83	10.00	9.98	9.83	10.34	10.10	9.83	10.00
GTRC	0.56	0.58	0.57	0.56	0.58	0.56	0.54	0.58	0.56	0.56	0.58	0.57

Table 1: Sensitivity of JCSA-RM with respect to the parameters $\{k, \kappa_p, th_b, th_d\}$.

We also compare the proposed method ***JCSA-RM*** (joint color-spatial-axial clustering and region merging) with several unsupervised RGB-D segmentation methods such as: RGB-D extension of OWT-UCM [25] (UCM-RGBD), modified Graph Based segmentation [9] with color-depth-normal (GBS-CDN), Geometry and Color Fusion method [8] (GCF) and the Scene Parsing Method [31] (SP). For the UCM-RGBD method we obtain best score with threshold value 0.1. The best results from GBS-CDN method are obtained by using $\sigma = 0.4$. To obtain the optimal multiplier (λ) in GCF [8] we exploit the range 0.5 to 2.5. For the SP method, we scaled the depth values (1/0.1 to 1/10 in meters) to use author's source code [31].

Table 2 presents (best appears as bold) the comparison w.r.t. the average score of the benchmarks. Results show that JCSA-RM performs best in PRI, VoI and GTRC and comparable in BDE and BFM. The reason is that, BDE and BFM favor methods like UCM-RGBD which is specialized in contour detection. This indicates that JCSA-RM can be improved by incorporating the boundary information more efficiently.

Several segmentation examples to visualize the results are illustrated in Fig. 2. We can see that the segmentation from JCSA-RM (our proposed) and UCM-RGBD are mostly competitive. However, they have several distinctions: (a) JCSA-RM is better in providing the detail of indoor scene structure whereas UCM-RGBD loose it sometimes (see ex. 3-5); (b) UCM-RGBD provides better estimation of the object boundaries whereas JCSA-RM gives a rough boundary and (c) UCM-RGBD shows more sensitivity on color whereas JCSA-RM

	PRI	VoI	BDE	GTRC	BFM
UCM-RGBD [25]	**0.90**	2.35	**9.11**	0.57	**0.63**
GBS-CDN [9]	0.81	2.32	13.23	0.49	0.53
GCF [8]	0.84	3.09	14.23	0.35	0.42
SP [31]	0.85	3.15	10.74	0.44	0.50
JCSA	0.87	2.72	10.33	0.45	0.46
JCSA-RM	**0.90**	**2.29**	9.83	**0.58**	0.59

Table 2: Comparison with the state of the art.

is more sensitive on directions. The GBS-CDN method provides visually pleasing results, however it often tends to loose details (see ex. 1-4) of the scene structure (e.g. merges wall with ceiling). Results from the SP method seem to be severely effected by the varying illumination and rough changes in surfaces (see ex. 3). The GCF method performs over-segmentation (see ex. 1, 3, and 5-7) or under-segmentation (see ex. 2 and 4), which is a drawback of such algorithms as they are often unable to estimate the correct number of clusters in real data. Moreover, the GCF method often fails to discriminate major surface orientations (see ex. 1, 2 and 4) as it does not consider the direction of surfaces (normal). Please see the supplementary material for additional results and analysis.

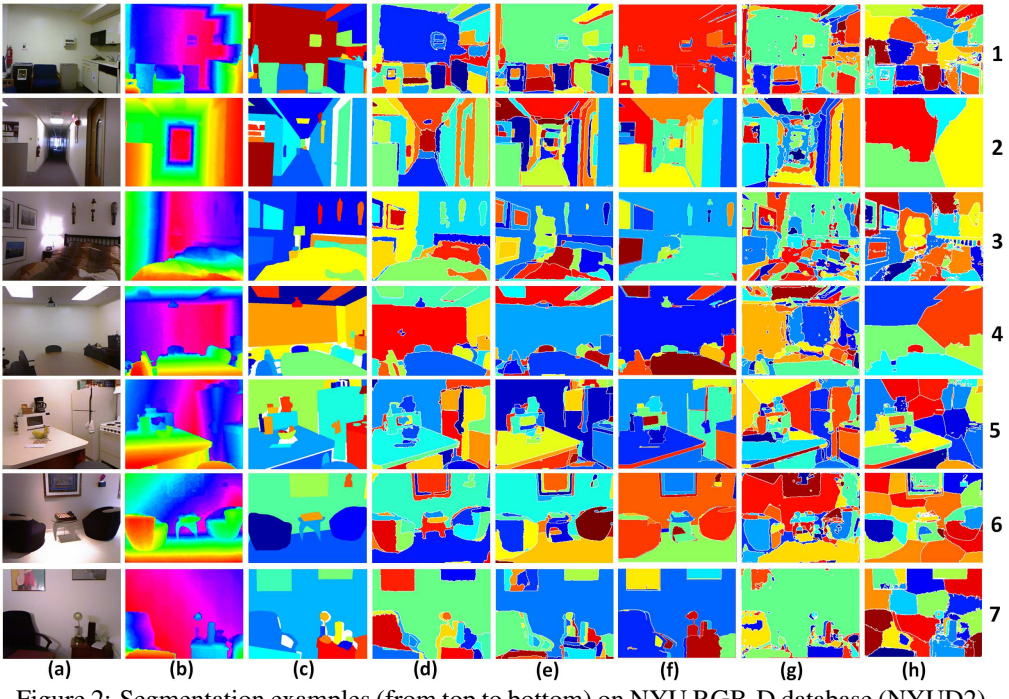

Figure 2: Segmentation examples (from top to bottom) on NYU RGB-D database (NYUD2). (a) Input Color image (b) Input Depth image (c) Ground truth (d) JCSA-RM (*our proposed*) (e) UCM-RGBD [25] (f) GBS-CDN [9] (g) SP [31] and (h) GCF [8].

Comparing JCSA with JCSA-RM (Table 2), we can decompose the contributions of *clustering* and *region merging* in JCSA-RM. We see that *region merging* improves clustering output from 0.45 to 0.58 (28.88%) in GTRC. We believe that JCSA-RM can be improved and extended further in the following ways:

- Including a pre-processing stage, which is necessary because the shape features are often computed inaccurately due to noise and quantization [5]. Moreover, we observed significant noise in the color images which are captured in the low light condition. A method like Scene-SIRFS [5] can be used for pre-processing purpose.
- Enhancing the clustering method by adding contour information [2] efficiently. Additionally, we may consider spatially constrained model such as [20].
- Enhancing the region merging method with color information. To this aim, we can exploit the estimated reflectance information (using [5]), such that the varying illumination is discounted.

In order to conduct the experiments we used a 64 bit machine with Intel Xenon CPU and 16 GB RAM. The JCSA-RM method is implemented in MATLAB, which on average takes 38 seconds, where 31 seconds for the clustering and 7 seconds for region merging. In contrast, UCM-RGBD (MATLAB and C++) takes 110 seconds. Therefore, JCSA-RM is ≈3 times faster[3] than UCM-RGBD. Moreover, we believe that implementing JCSA-RM in C++ will significantly reduce the computation time.

To further analyze the computation time of JCSA-RM, we run it for different image scales. Table 3 presents relevant information from which we see that the reduction rate of JCSA computation time (in sec) w.r.t. different scales is approximately equivalent to the reduction rate of the number of pixels.

Scale	**1**	**1/2**	**1/4**	**1/8**
Num. pixels	239k	60k	15k	4k
JCSA (req. time in sec)	132	31	8	1.5
RM (req. time in sec)	42	7	1.4	0.33

Table 3: Computation time of JCSA-RM w.r.t. different image scales.

4 Conclusion

We propose an unsupervised indoor RGB-D scene segmentation method. Our method is based on a statistical image generation model, which provides a theoretical basis for fusing different cues (e.g. color and depth) of an image. In order to cluster w.r.t. the image model, we developed an efficient joint color-spatial-axial clustering method based on Bregman Divergence. Additionally, we propose a region merging method that exploits the planar statistics of the image regions. We evaluate the proposed method with a benchmark RGB-D image database and using widely accepted evaluation metrics. Results show that our method is competitive w.r.t. the state of the art and opens interesting perspectives for fusing color and geometry. We foresee several possible extensions of our method: more complex image model and clustering with additional features, region merging with additional hypothesis based on color. Moreover, we believe that the methodology proposed in this paper is equally applicable and extendable for other complex tasks, such as joint image-speech data analysis.

Acknowledgment

This work has been supported by a research grant from the ARC6 of région Rhône-Alpes, France.

[3] To perform a fair comparison, we conducted this experiment with half scaled image. This is due to the fact that the computational resource did not support to run UCM-RGBD for the full scale image.

References

[1] Olivier Alata and Ludovic Quintard. Is there a best color space for color image characterization or representation based on multivariate gaussian mixture model? *Computer Vision and Image Understanding*, 113(8):867–877, 2009.

[2] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):898–916, 2011.

[3] David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics, 2007.

[4] Arindam Banerjee, Srujana Merugu, Inderjit S Dhillon, and Joydeep Ghosh. Clustering with bregman divergences. *The Journal of Machine Learning Research*, 6:1705–1749, 2005.

[5] Jonathan T Barron and Jitendra Malik. Intrinsic scene properties from a single rgb-d image. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17–24. IEEE, 2013.

[6] Christophe Biernacki, Gilles Celeux, and Gérard Govaert. Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(7):719–725, 2000.

[7] Ming-Ming Cheng, Guo-Xin Zhang, Niloy J. Mitra, Xiaolei Huang, and Shi-Min Hu. Global contrast based salient region detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 409–416, 2011.

[8] Carlo Dal Mutto, Pietro Zanuttigh, and Guido M Cortelazzo. Fusion of geometry and color information for scene segmentation. *IEEE Journal of Selected Topics in Signal Processing*, 6(5):505–521, 2012.

[9] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.

[10] Jordi Freixenet, Xavier Muñoz, David Raba, Joan Martí, and Xavier Cufí. Yet another survey on image segmentation: Region and boundary information integration. In *Computer Vision-ECCV 2002*, pages 408–422. Springer, 2002.

[11] Vincent Garcia and Frank Nielsen. Simplification and hierarchical representations of mixtures of exponential families. *Signal Processing*, 90(12):3197–3212, 2010.

[12] Saurabh Gupta, Pablo Arbelaez, and Jitendra Malik. Perceptual organization and recognition of indoor scenes from rgb-d images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 564–571. IEEE, 2013.

[13] Md Abul Hasnat, Olivier Alata, and Alain Trémeau. Hierarchical 3-d von mises-fisher mixture model. In *1st Workshop on Divergences and Divergence Learning (WDDL)*, 2013.

[14] Md. Abul Hasnat, Olivier Alata, and Alain Trémeau. Unsupervised clustering of depth images using watson mixture model. In *International Conference on Pattern Recognition (ICPR)*. IEEE, 2014.

[15] Hema Swetha Koppula, Abhishek Anand, Thorsten Joachims, and Ashutosh Saxena. Semantic labeling of 3d point clouds for indoor scenes. In *NIPS*, volume 1, page 4, 2011.

[16] Yi Ma, Harm Derksen, Wei Hong, and John Wright. Segmentation of multivariate mixed data via lossy data coding and compression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(9):1546–1562, 2007.

[17] Kanti V Mardia and Peter E Jupp. *Directional statistics*, volume 494. Wiley. com, 2009.

[18] Adolfo Martínez-Usó, Filiberto Pla, and Pedro García-Sevilla. Unsupervised colour image segmentation by low-level perceptual grouping. *Pattern Analysis and Applications*, 16(4):581–594, 2013.

[19] Kevin P Murphy. *Machine learning: a probabilistic perspective*. The MIT Press, 2012.

[20] Thanh Minh Nguyen and Qm Wu. Fast and robust spatially constrained gaussian mixture model for image segmentation. *IEEE transactions on circuits and systems for video technology*, 23(4):621–635, 2013.

[21] Frank Nielsen and Vincent Garcia. Statistical exponential families: A digest with flash cards. *CoRR*, abs/0911.4863:http://arxiv.org/abs/0911.4863v2, 2011.

[22] Yuzhen Niu, Yujie Geng, Xueqing Li, and Feng Liu. Leveraging stereopsis for saliency analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 454–461. IEEE, 2012.

[23] Richard Nock and Frank Nielsen. Statistical region merging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(11):1452–1458, 2004.

[24] Bo Peng and David Zhang. Automatic image segmentation by dynamic region merging. *IEEE Transactions on Image Processing*, 20(12):3592–3605, 2011.

[25] Xiaofeng Ren, Liefeng Bo, and Dieter Fox. Rgb-(d) scene labeling: Features and algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2759–2766. IEEE, 2012.

[26] Radu Bogdan Rusu. *Semantic 3D Object Maps for Everyday Robot Manipulation*. Springer, 2013.

[27] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *Computer Vision–ECCV 2012*, pages 746–760. Springer, 2012.

[28] Suvrit Sra and Dmitrii Karp. The multivariate watson distribution: Maximum-likelihood estimation and other aspects. *J Multivar Anal*, 114:256 – 269, 2013.

[29] Johannes Strom, Andrew Richardson, and Edwin Olson. Graph-based segmentation for colored 3d laser point clouds. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2131–2136. IEEE, 2010.

[30] Richard Szeliski. *Computer vision: algorithms and applications*. Springer, 2011.

[31] Camillo J Taylor and Anthony Cowley. Parsing indoor scenes using rgb-d imagery. *Robotics: Science and Systems VIII*, pages 401–408, 2013.

[32] Alain Trémeau and Philippe Colantoni. Regions adjacency graph applied to color image segmentation. *IEEE Transactions on Image Processing*, 9(4):735–744, 2000.