Estimating Floor Regions in Cluttered Indoor Scenes from First Person Camera View

Sanchit Aggarwal, Anoop M. Namboodiri, C. V. Jawahar
CVIT, International Institute of Information Technology, Hyderabad, India
{sanchit.aggarwal@research. , anoop@, jawahar@}iiit.ac.in

***Abstract*—The ability to detect floor regions from an image enables a variety of applications such as indoor scene understanding, mobility assessment, robot navigation, path planning and surveillance. In this work, we propose a framework for estimating floor regions in cluttered indoor environments. The problem of floor detection and segmentation is challenging in situations where floor and non-floor regions have similar appearances. It is even harder to segment floor regions when clutter, specular reflections, shadows and textured floors are present within the scene. Our framework utilizes a generic classifier trained from appearance cues as well as floor density estimates, both trained from a variety of indoor images. The results of the classifier is then adapted to a specific test image where we integrate appearance, position and geometric cues in an iterative framework. A Markov Random Field framework is used to integrate the cues to segment floor regions. In contrast to previous settings that relied on optical flow, depth sensors or multiple images in a calibrated setup, our method can work on a single image. It is also more flexible as we avoid assumptions like Manhattan world scene or restricting clutter only to wall-floor boundaries. Experimental results on the public MIT Scene dataset as well as a more challenging dataset that we acquired, demonstrate the robustness and efficiency of our framework on the above mentioned complex situations.**

Keywords*—*Scene Understanding, Floor Segmentation, Cluttered Indoor Scenes.

I. Introduction

One of the classical research problems in computer vision is the segmentation of a given natural scene image into semantically meaningful entities. This forms the basis of a variety of tasks such as navigation, free space estimation, image enhancement, 3D reconstruction, scene understanding and classification [1]–[4]. Restricting the problem to indoor scenes makes the problem more tractable, and researchers have met with varying success in recent years. Some of these approaches employ advanced sensing methods such as active cameras (Kinect) and, calibrated multi-image setups to extract depth data [2]. Others make strong assumptions about the camera and world such as smooth camera motion parallel to the ground plane with no rotation and roll [1] or Manhattan world assumptions [3]. Detection of floor from a single indoor image is still an open problem, especially when the scene is cluttered with objects like furnitures or if the floors and walls are textured with tiles, panels and curtains. It is even more difficult to detect floor regions in scenarios where shadows and highlights are present due to illumination or when clutter is not confined to wall-floor boundaries. Another challenge is scenes with non-Manhattan layout where adjacent walls are not perpendicular to each other. In this paper we propose an efficient and reliable framework for floor detection that works in presence of challenging situations demonstrated in Figure 1.

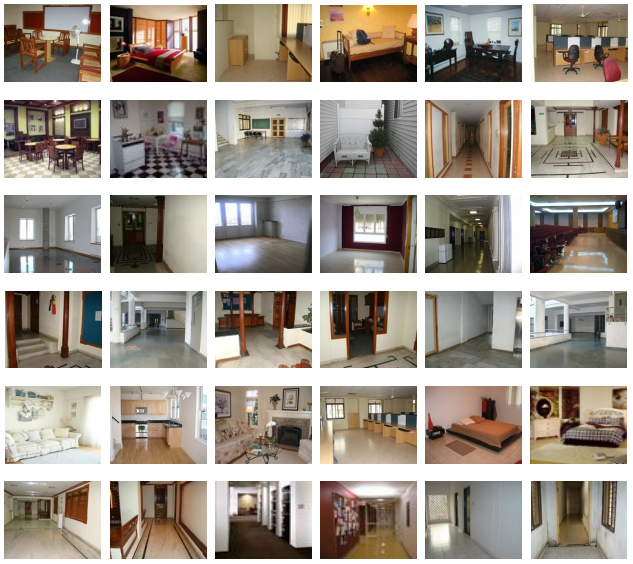

Fig. 1: Sample images from dataset showing *challenging situations* in indoor scenes: Rows 1) Clutter not confined to floor boundaries, 2) Varied floor texture, 3) Specular reflections and shadows, 4) Scenes not satisfying box layout [3] or Manhattan assumptions [5], 5) Similar floor and non-floor regions, and 6) Long corridors with varied perspective. (Best viewed in color)

We now take a closer look at the more recent work on floor estimation and scene understanding from single or multiple images [1]–[3]. These methods are quite powerful although based on strong assumptions or constraints such as non-occluded wall-floor boundaries [6], need for depth data for semantic understanding [2], [7], or need for smooth motion for accurate tracking [1] or optical flow. These attempts cover only limited scenarios, and cannot be generalized to a wide variety of practical cases. Hedau *et al.* [3] approached the problem of recovering spatial layout of indoor scenes from monocular images by modeling the global room space with a parametric 3D box with Manhattan world assumptions. Tsai *et al.* [1] created a model for three-wall indoor environment with no occluding edges from the experience of artificial agent mounted with a fixed camera and moving parallel to ground plane. Zhang *et al.* [2] jointly estimates the layout of the rooms and the clutter present in the scene using RGB-D data. Delage *et al.* [8] presented a dynamic Bayesian network model for recovering 3D information for many images with long and non-occluded floor-wall geometry. These methods generally fail in many real-world scenarios where it is not necessary to have non-occluded or clear edges. Some of the methods also require specialized sensors for accurate results.

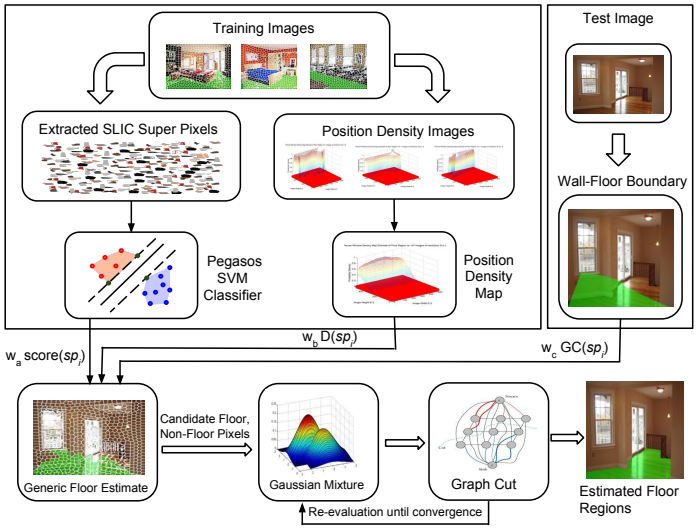

Fig. 2: Overview of the proposed algorithm. Labeled superpixels from training data are used to learn a SVM classifier and a floor likelihood map. Given a test image, a weighted sum of SVM scores, position densities and wall-floor boundary cues are use to compute a generic floor estimate. This is then use to generate candidate floor and non-floor pixels and GMMs estimates are formed for floor and non-floor regions. A MRF over the GMM log likelihoods is optimized by max flow/min cut to obtain final floor regions. (Best viewed in color)

We would like to estimate the floor in complex scenarios without resorting to assumptions such as Manhattan world layout, clear visibility of three walls or non-occluding edges and avoid the need for calibrated camera pairs. The goal is to have a robust approach that can adapt to a specific scenario and work with a single monocular camera. We would also like to detect floor regions in natural settings where there are shadows from objects and strong specular reflections. We also intend to make the algorithm efficient enough for practical scenarios with minimal constraints as mentioned before.

The main contribution of our work is a robust framework that can estimate floor regions in the above mentioned complex scenarios. We attain this robustness with effective integration of the position, appearance and geometric cues learned from the images, and by iterative re-evaluation of these cues that adapts the model to a given test image. Analysis of our experiments over a set of 460 images of varying complexity shows the effectiveness of our methodology. In Section II, we give the overview of our framework and describe the dataset and features used for various models. Section III describes about the inferencing technique followed by results and analysis of experiments in Section IV.

II. Proposed Framework

An overview of our framework is illustrated in Figure 2. We propose a method for combining generic cues i.e the most likely floor appearance (Sec. II-A1) and position estimates (Sec. II-A2) learned from set of images with image specific appearance and geometry cues (Sec. II-A3) to estimate the accurate floor regions in the indoor image. We impose an implicit constraint on the scene, having floor region near to the camera (which was automatically learned from training images). We start by learning a Support Vector Machine (SVM) [9] classifier for generic floor appearance and floor position cues based on a Parzen-window density estimates across the set of images. We use them for estimating possible regions of floor in a given test image. We then compute image specific appearance cues based on a Gaussian Mixture Models (GMM) and floor geometry cues based on extracted floor polygons. A Maximum a Posteriori (MAP) estimate for the Markov random field (MRF) with the above cues is computed on the set of SLIC superpixels [10] computed from the image. An iterative minimization scheme [11] is used for segmenting floor and non-floor regions by re-evaluation of the specific appearance cue and optimization using graphcut [12]. This approach of re-learning of appearance cues results in effective and fast estimation of the floor regions.

A. *Evidence Integration for Floor Segmentation*

Given an indoor scene, there are two cues that help us in discerning the floor regions from walls and furnitures. (i) The floor regions usually have different color and texture compared to walls and furnitures. In most of the indoor scenes, walls are more likely to be plain while floors are regularly textured, and there is a contrast difference between walls and floors. (ii) In few cases this difference may not be obvious due to presence of clutter with the same texture as of floors, variation of texture within the floors, or shadows from clutter and low ambient illumination. However, in complex images like these observed from the first-person view, the bottom region of the image is more likely to be a floor as compared to the top regions. We exploit these hidden differences between floor and non-floor regions by supervised learning approach and Parzen-window density estimation for generic appearance and position cues respectively. We then estimate a wall-floor polygon to compute geometric cues in the given test image. We use the generic classifiers in adittion with the geometric cues to get the rough estimate of generic floor region. The generic floor estimate is then smoothened using specific apearance cues to get the final estimate of floor in an image.

*1) **Generic Appearance Cues**:* Given a set of images, we over-segment each image into superpixels using simple linear iterative clustering (SLIC) [10]. We use SLIC as it is fast to compute and it produces superpixels of regular sizes which adhere to wall-floor boundaries within the image. Other methods like Felzenszwalb and Huttenlocher's graph-based approach [13] produces superpixels of irregular shapes which are not suitable for our method. A feature vector of 134-dimension is extracted for each superpixel based on its appearance similar to Hoeim *et al.* [14] with following modifications.

- *Color Cues (30-dim)*: RGB/HSV mean(6) and Hue, Saturation and Value Histogram ($16 + 4 + 4$ bins respectively).
- *Location and Shape Cues (8-dim)*: normalized x, y, 10th, 50th and 90th percentile, area and eccentricity of superpixel.
- *Texture Cues (96-dim)*: mean absolute response and histogram of maximum responses using 48 Leung-Malik(LM) filter bank [15].

A hyperplane w, separating the two classes of floor and non-floor superpixels is learned using Primal Estimated sub-GrAdient SOlver for SVM (PEGASOS) [16] which is fast and effective and provides a model for online training. Dimensions of the input feature vector is increased by 3 times by using χ^2 explicit feature map of [17] for non-linear additive kernels. Regularization-loss trade-off parameter C of SVM is determined by 3-Fold Cross validation over different value of C for $T = 1000$ where T is number of iterations.

The learned SVM model is used to compute scores for superpixel sp_i in a given test image. The SVM model was calibrated to return posterior probabilities using the sigmoid fitting method described in [18]. For example, floor probability scores for superpixel sp_i takes the form:

$$score\left(sp_i\right) = \left[1 + \exp\left(Af + B\right)\right]^{-1}, \tag{1}$$

where $f = f(sp_i)$ is score returned by SVM model and A,B are Platt scaling parameters [18].

2) ***Generic Position Density Map****:* Since floor region occurs in specific positions within a scene, learning the underlying probability density function (PDF) for position of floor pixels can be a discriminative cue for estimating floor regions in the image. The most effective non-parametric method to learn this underlying probability distribution is kernel density estimation (KDE) approach also termed as Parzen-window method [19]. In this section we describe the steps for the computation of probability density image D_i, for given indoor scene with groundtruth floor mask M_i, defined as :

$$M_i(x, y) \equiv I = \begin{cases} 1 & \text{if Floor Pixel} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Let $Z \equiv (x, y)$ be the set of all pixels locations of I. Therefore $X \equiv (Z|I = 1)$ is a set of all floor pixel locations. Then the bivariate normal density for a given floor pixel sample k at location X_k is :

$$p_N\left(X_k\right) = \frac{1}{N}\sum_{j=1}^{N}\frac{1}{h_N}\varphi\left(\frac{X_k - Z_j}{h_N}\right) \tag{3}$$

where $N = (m \times n)$ is the size of mask M_i, h_N is bandwidth parameter and $\varphi(x) = \frac{1}{\sqrt{2\pi}}\mathrm{e}^{-x^2/2}$ is a bivariate normal density kernel with zero mean and unit variance. The probability density image D_i is average of normal densities $\forall$ k $\in X$ given by :

$$D_i = \frac{1}{N}\sum_{k=1}^{|X|} p_N\left(X_k\right) \tag{4}$$

Given T such training mask images, normalized position density map is estimated from the average of density images :

$$\hat{D} = \frac{1}{T}\sum_{t=1}^{T} D_t \tag{5}$$

The decision regions for KDE depend upon the choice of bandwidth parameter h_N. For efficient and automatic bandwidth selection we used the implementation of adaptive kernel density based on linear diffusion process as proposed in [20] for estimating D_i. An adaptive kernel is constructed by considering the most general linear diffusion with its stationary density equal to a pilot density estimate. This allows fast computation of the underlying pdf and results in a diffusion estimator which is consistent at boundaries.

3) ***Test Image specific Geometry Cues****:* In images with similar appearances additional geometry cues are needed to differentiate superpixels belonging to floor and non-floor regions. These cues can be computed from the rough estimation of the floor-wall boundary. For fast and efficient estimation of this wall-floor boundary we use the modified version of the algorithm proposed in Li *et al.* [6]. We extract long straight lines (*minimum length* > 70) with the method used by [21], and divide them into three categories of horizontal lines, vertical lines and inclined lines with corresponding slopes between 0-10, 20-65 and 85-90 degrees respectively. We then compute all the intersection points for inclined lines. We divide the y coordinates of the intersection points into equal bins of size 10 and selected bin with highest frequency. We then calculate the mean (x, y) for all points belonging to this bin and remove all intersection points and their contributing lines which are above this. We again calculate the mean (x, y) of the remaining intersection points.

This method of pruning the intersection points results in the most likely candidate for vanishing point. We remove all the horizontal lines and vertical lines above this vanishing point. Similar to Li *et al.* [6], we then estimate the rough boundary by connecting the endpoints of each vertical lines and computed a bottom score for inclined lines. We select all inclined lines above the threshold to estimate a approximate wall floor boundary polygon. Based on the detected wall-floor boundary, geometric cues are computed for each superpixel. Geometric Cue is region of superpixel sp_i inside the boundary and is given by :

$$GC\left(sp_i\right) = \frac{\text{No of pixels of } sp_i \text{ inside wall-floor boundary}}{\text{Total pixels in } sp_i} \tag{6}$$

4) ***Estimating Generic Floor Mask****:* Weighted sum of svm scores, geometric cues and position density map is computed for each superpixel sp_i :

$$generic\left(sp_i\right) = w_a score\left(sp_i\right) + w_b\hat{D}\left(sp_i\right) + w_c GC\left(sp_i\right) \tag{7}$$

where w_a, w_b, w_c are the normalized weights. They are learned experimentally from the individual prediction accuracies of generic appearance cues, position density map and geometric cues respectively. They are set to 0.29, 0.37 and 0.34 respectively for our experiments. We use Generic scores to estimate the *generic floor mask* for a given test image of indoor environment.

5) ***Specific Appearance Cues****:* We use Gaussian Mixture Models (GMM) to represent the candidate pixels for floor and non-floor region estimated from the *generic floor mask*. Due to the generative nature of GMM, they are good in representing and learning large class of appearance distribution which might have been missed during the discriminative learning process of SVM for generic appearance cues. In case of floor regions estimation from a test image, individual components of GMM can efficiently model underlying appearance differences within floor and non-floor regions and are useful for their smooth approximations. We use GMM with $K = 5$ components and initialized them for both floor and non-floors sets using 5-dimension feature vector Z, consisting R,G,B,X,Y values of pixels instead of just R,G,B as used by [12]. The posterior

probability score of pixel p can be computed from floor and non-floor GMM as follows :

$$P(p) = -log \sum_{k=1}^{K} \pi_k \frac{1}{\sqrt{|\Sigma_k|}} e^{-\frac{1}{2}[Z_p - \mu_k]^T \Sigma_k^{-1} [Z_p - \mu_k]} \quad (8)$$

The above probabilities scores are used for smoothing the generic floor estimate by setting a Markov random field.

III. Iterative MAP-MRF Inference

We use the Grabcut algorithm proposed in [12] for our final estimation of floor regions from generic floor mask obtained from position density map, generic appearance cues and geometric cues. We setup a Markov random field for the binary classification problem of estimating possible floor regions within an indoor image and assigning unique class label $(\alpha_i = 1/0)$ for floor/non-floor set of pixels $p = (p_1, p_2, \ldots, p_k)$. The solution $\alpha = (\alpha_1, \alpha_2, ..., \alpha_k)$ occurs at the maxima of posterior probability given by :

$$p(\alpha|p) = \frac{1}{Z(p)} e^{-E(\alpha,p)} \quad (9)$$

where $E(\alpha, p)$ is the Gibbs energy function and $Z(p)$ is the normalizing factor. The gibbs energy function is often expressed as the sum of data term and smoothness term

$$E(\alpha, p) = U(\alpha, p) + V(\alpha, p). \quad (10)$$

The data term $U(\alpha, p)$ for pixels are scores from specific appearance cues and is defined as

$$U(\alpha, p) = P(p). \quad (11)$$

Similar to [11], smoothness term is defined as the pairwise penalty due to link between pixels p_m and p_n

$$V(\alpha, p) = \frac{\gamma}{dist(m,n)} e^{-\beta \|Z_m - Z_n\|^2} \quad (12)$$

where Z_m and Z_n are 5-dimension specific appearance cues for pixel p_m and p_n respectively . As proposed [22], the constant β is set to $\beta = \left(2\langle Z_m - Z_n \rangle^2\right)^{-1}$ and $\gamma = 50$. We use the max flow-min cut algorithm of graphcut [12] for global optimization of flow network where the nodes weights are represented by data term $(U(\alpha, p))$ and edge weights are represented by smoothness term $(V(\alpha, p))$. Similar to grabcut algorithm [11], we initialize trimap T from the two candidate sets of floor and non-floor pixels estimated from the generic floor mask. T_B is set of all the pixels in non-floor candidate set above confidence ϕ, $T_F = \varnothing$ and $T_U = \overline{T_B} \cup \{CandidateFloorPixels\}$. We set the confidence $\phi = 90\%$ for our experimentation. We re-evaluate the specific appearance cues for iterative minimization.

IV. Experiments and Results

A. Dataset and Evaluation Metrics

For the experiments, we collect 350 images from public MIT scene dataset that were indoor scenes with visible floor regions. In addition we capture 110 images from various buildings in our campus that include cluttered floor regions. The images contain wide variety of indoor scene including classrooms, living rooms, library, corridors, three or two visible walls, etc. We divide the images from the two datasets into three groups. The first is referred to as the *Clutter Dataset*, and consists of images with varied texture within the floor, specular highlights, shadows and scenes with cluttered floors due to furnitures or other obstacles, where the clutter is not just confining to image boundaries. The *Non-Clutter Dataset* contains corridor like images with minimal clutter and large floor regions. A *Mixed Dataset* was also defined that contains both cluttered and non-cluttered scenes for the purpose of evaluating performance in real-world applications. All the images were manually annotated with floor and non-floor regions using the web-based image annotation tool LabelMe [23]. We randomly divide the images into training and testing within the three sets and learn SVM and floor position densities for all the three sets respectively. All experiments described in this section have been performed on the images resized to resolution of 512×512.

TABLE II: Average computational time per image required by our approach and that of Hedau *et al.* [3]. The proposed method is significantly faster than [3], and could be used in practical applications with an optimized implementation.

MIT Scene Dataset	No. of Test Images	**Our Approach**	Hedau *et al.* [3]
Clutter	72	**11.31 s**	274.96 s
Non-Clutter	34	**14.18 s**	556.45 s
Mixed	105	**10.98 s**	268.91 s
Our Dataset	No. of Test Images	**Our Approach**	Hedau *et al.* [3]
Clutter	15	**17.58 s**	232.29 s
Non-Clutter	18	**18.40 s**	269.60 s
Mixed	33	**15.92 s**	429.41 s

In the binary classification problem of floor segmentation where the two classes of floor and non-floor pixels in cluttered indoor image are highly imbalanced, traditional performance evaluation metric based on classification accuracy is not the most appropriate one. Due to the overriding presence of non-floor regions in highly cluttered images we selected the geometric mean or G-Mean, as suggested by [24], as the metric of performance evaluation for the proposed framework. In binary classification, sensitivity and specificity are true positive rate and true negative rate respectively and are given by

$$Sensitivity = \frac{TruePositive}{TruePositive + FalseNegative} \quad (13)$$

$$Specificity = \frac{TrueNegative}{FalsePositive + TrueNegative} \quad (14)$$

G-Mean is then defined as

$$\text{G-Mean} = \sqrt{\text{Sensititvity} \times \text{Specificity}} \quad (15)$$

B. Results and Discussions

We first individually evaluate the generic appearance cues, generic position cues and specific geometry cues which are then used to evaluate the specific appearance cues. Figure 3 shows the results of floor segmentation after each steps of our approach for 10 different indoor scenes with various challenging situations for the three sets of cluttered, non-cluttered and mixed datasets. Table I shows a quantitative comparison of our approach with the state-of-the-art methods suggested in [3]. We use the same evaluation criteria for all comparisons. Table II shows the computation time required by our framework and that of Hedau *et al.* [3]. As seen from Table I, the proposed algorithm achieves a better G-Mean score

TABLE I: Accuracy and G-Mean for Cluttered, Non-Cluttered and Mixed Dataset. Columns(1-10): Accuracy and G-Mean after each step of our framework. Baseline: Results with the spatial layout framework by Hedau *et al.* [3]

	Accuracy and G-mean each step for proposed framework										**Baseline**	
	1. SVM		2. KDE		3. BOUNDARY		1+2+3		**Final**		**Hedau *et al.* [3]**	
MIT Scene Dataset	Accuracy	G-Mean	Accuracy	G-Mean	Accuracy	G-Mean	Accuracy	G-Mean	**Accuracy**	**G-Mean**	**Accuracy**	**G-Mean**
Clutter	83.60	87.63	88.18	82.45	84.47	68.76	87.15	78.02	89.23	**88.04**	**90.12**	80.58
Non-Clutter	90.92	78.67	89.54	83.63	89.23	76.34	91.35	85.18	**94.41**	**94.96**	92.91	85.71
Mixed	88.75	83.24	87.36	79.25	85.32	69.61	87.58	77.74	90.08	**87.85**	**90.45**	81.02
	1. SVM		2. KDE		3. BOUNDARY		1+2+3		**Final**		**Hedau *et al.* [3]**	
Our Dataset	Accuracy	G-Mean	Accuracy	G-Mean	Accuracy	G-Mean	Accuracy	G-Mean	**Accuracy**	**G-Mean**	**Accuracy**	**G-Mean**
Clutter	90.13	73.43	85.56	73.72	85.10	68.53	88.60	78.15	91.20	**89.48**	**93.99**	78.47
Non-Clutter	91.91	91.17	86.17	86.65	85.71	78.25	90.51	89.03	**92.00**	**91.37**	82.32	71.47
Mixed	90.59	82.03	85.52	83.28	85.43	71.85	89.23	83.67	**90.81**	**88.73**	84.95	76.39

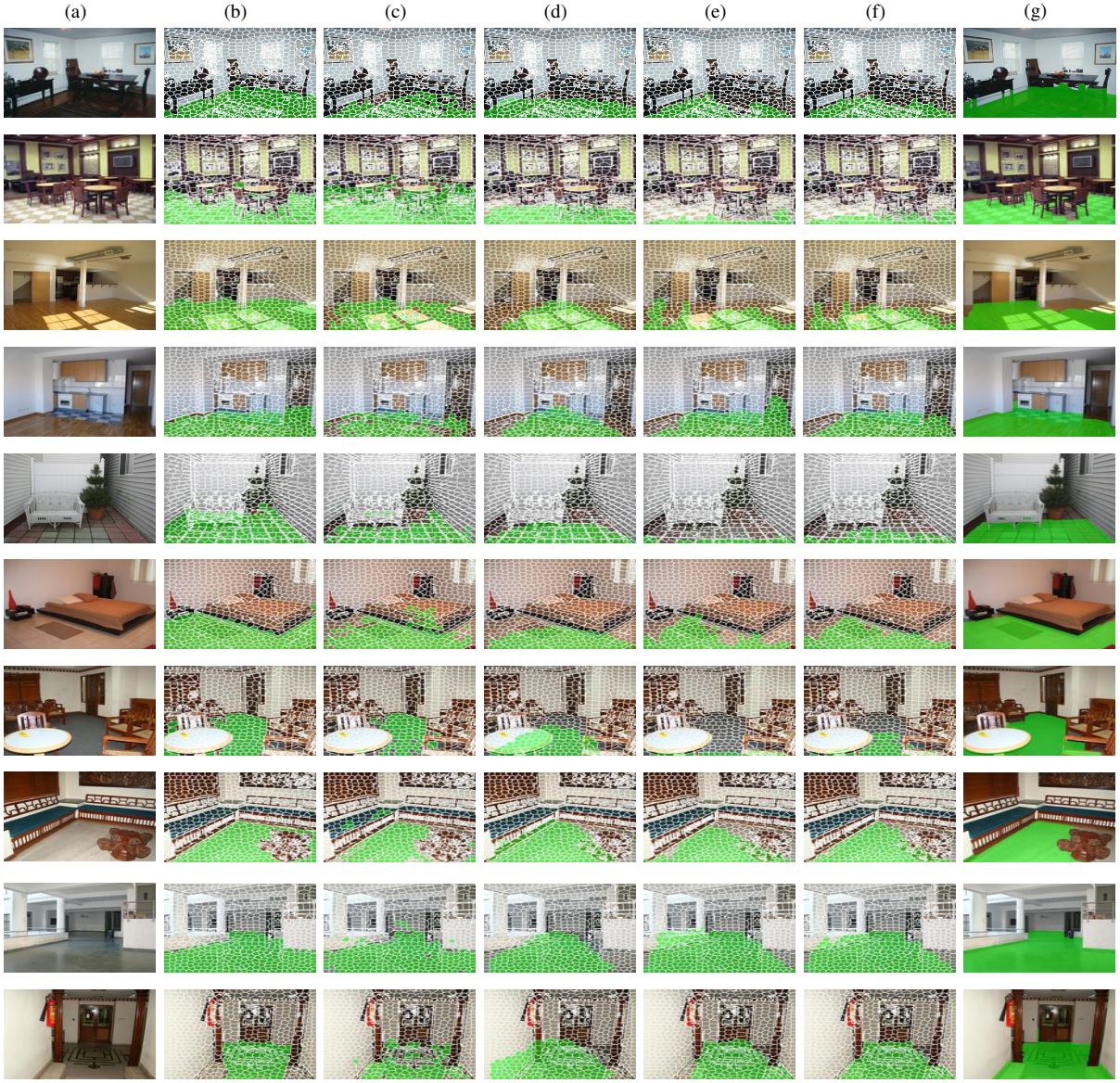

Fig. 3: Estimated Floor regions for each step of the proposed Framework on MIT (Rows 1-6) and Our (Rows 7-10) datasets. Columns (c-g) Segmented floor regions after every step, (a) Original Image, (b) Superpixel GroundTruth, (c) SVM Classification, (d) Kernel density estimate, (e) Boundary Detection, (f) Generic (SVM+KDE+Boundary Floor), (g) Final segmented floor regions. Results for SVM and KDE trained on three different sets of clutter, non-clutter and mixed training images are shown in (Row 1,2,7), (Row 3,4,8) and (Row 5,6,9,10) respectively.(Best viewed in color)

compared to [3] in all the cases. The classification accuracy, while not the best indicator of performance, is also better than [3] in most cases. The reason is primarily due to the removal of assumptions about the nature of walls in the room, which is not satisfied in most practical scenarios.

The important point to note is that our method achieve improvement in performance along with the significant reduction in computational requirements compared to [3] as seen from Table II. This is because of using relatively simple models for appearance, geometric and generic cues at each step of evidence integration. The computational time is further reduced due to inference on the SLIC superpixels. Figure 4 shows the results of our algorithm where the traditional methods of fitting a box layout fails due to the non-Manhattan world structure present in indoor scenes. Some examples where the algorithm fails are shown in Figure 4, Row 5. From left to right, the floor area moves farther from the camera and clutter near the bottom of the image increases making it overly clutter i.e more than 95% of the image area belongs to non-floor region, due to which the algorithm mistakenly predict non-floor region as floor regions.

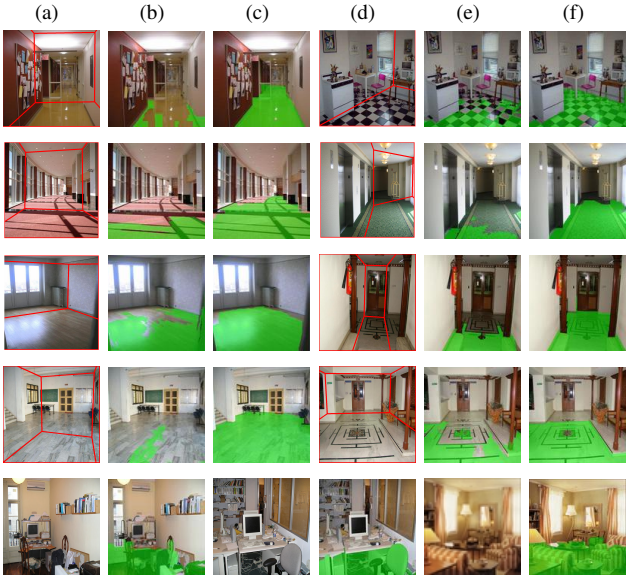

Fig. 4: Rows 1-4) Column *c* and *f* shows the results of our framework on scenarios that violates the Manhattan assumptions, or have specular reflection and varied texture. Columns *a* and *d* are the box layout detected by Headu *et al.* [3] resulting in the surface labels given in columns *b* and *e*. Detection of box layout based on Manhattan assumption fails in indoor scenes having walls that are not orthogonal to each other. Rows 5, shows three failure cases of our algorithm when images are densely clutter with furniture and floor region is farther from the camera. (Best viewed in color)

V. Conclusions and Future Work

In this work, we have proposed a robust approach for segmenting floor pixels in complex indoor scenes that effectively combines cues developed for generic indoor scenes with the cues learned from the specific test image. Our experimental results show that the generic cues and specific cues allow efficient and effective estimation by giving support to each other and the algorithm is capable of working even in situations where there are shadows, specular highlights and with varied appearance of floor texture. The approach is also not constrained to camera motion, which makes it easy to use in situations where camera motion is erratic as in first person view applications. We would like to extend the proposed work for videos where a person wearing a camera is able to detect walkable floor regions which might be helpful in navigation for the visually challenged.

References

[1] G. Tsai, C. Xu, J. Liu, and B. Kuipers, "Real-time indoor scene understanding using bayesian filtering with motion cues," in *Proc. ICCV*, 2011.

[2] J. Zhang, K. Chen, A. G. Schwing, and R. Urtasun, "Estimating the 3D Layout of Indoor Scenes and its Clutter from Depth Sensors," in *Proc. ICCV*, 2013.

[3] V. Hedau, D. Hoiem, and D. Forsyth, "Recovering the spatial layout of cluttered rooms," in *Proc. ICCV*, 2009.

[4] M. Juneja, A. Vedaldi, C. Jawahar, and A. Zisserman, "Blocks that shout: Distinctive parts for scene classification," in *Proc. CVPR*, 2013.

[5] D. C. Lee, M. Hebert, and T. Kanade, "Geometric reasoning for single image structure recovery," in *Proc. CVPR*, 2009.

[6] Y. Li and S. T. Birchfield, "Image-based segmentation of indoor corridor floors for a mobile robot," in *Proc. IROS*, 2010.

[7] N. Silberman and R. Fergus, "Indoor scene segmentation using a structured light sensor," in *Proc. ICCV*, 2011.

[8] E. Delage, H. Lee, and A. Y. Ng, "A Dynamic Bayesian Network Model for Autonomous 3D Reconstruction from a Single Indoor Image," in *Proc. CVPR*, 2006.

[9] V. Vapnik, "An overview of statistical learning theory," *IEEE Transactions on Neural Networks*, 1999.

[10] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Ssstrunk, "SLIC Superpixels Compared to State-of-the-art Superpixel Methods," *In PAMI*, 2012.

[11] V. K. Carsten Rother and A. Blake, "Grabcut-interactive foreground extraction using iterated graph cuts," *ACM TRANS. GRAPH*, 2004.

[12] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *In PAMI*, 2001.

[13] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *IJCV*, 2004.

[14] A. A. E. Derek Hoiem and M. Hebert, "Recovering surface layout from an image," *IJCV*, 2007.

[15] T. Leung and J. Malik, "Representing and recognizing the visual appearance of materials using three-dimensional textons," *IJCV*, 2001.

[16] Y. Singer and N. Srebro, "Pegasos: Primal estimated sub-gradient solver for svm," in *In ICML*, 2007.

[17] A. Vedaldi and A. Zisserman, "Efficient additive kernels via explicit feature maps," *In PAMI*, 2012.

[18] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *ADVANCES IN LARGE MARGIN CLASSIFIERS*, 1999.

[19] E. Parzen, "On estimation of a probability density function and mode," *The Annals of Mathematical Statistics*, 1962.

[20] Z. Botev, J. Grotowski, D. Kroese *et al.*, "Kernel density estimation via diffusion," *The Annals of Statistics*, 2010.

[21] J. Kosecka and W. Zhang, "Video compass," in *Proc. ECCV*, 2002.

[22] Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images," in *Proc. ICCV*.

[23] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "Labelme: A database and web-based tool for image annotation," *IJCV*, 2008.

[24] M. Kubat and S. Matwin, "Addressing the curse of imbalanced training sets: One-sided selection," in *Proc. ICML*, 1997.