Proceedings of the 5th WSEAS International Conference on Telecommunications and Informatics, Istanbul, Turkey, May 27-29, 2006 (pp366-370)

# A Learning-based Algorithm for Geometric Labeling of Indoor Images

XIAOQING LIU
The University of Western Ontario
Dept. of Electrical & Computer Engineering
London ON N6A 5B9
CANADA

JAGATH SAMARABANDU
The University of Western Ontario
Dept. of Electrical & Computer Engineering
London ON N6A 5B9
CANADA

*Abstract:* This paper aims to use a large set of feature descriptions as geometric cues to build the structural knowledge of an indoor image. In this paper, a large quantity of training images are used to obtain the required information through learning. We apply a multi-class version of AdaBoost with weak learners based on the decision tree to label regions in an indoor image as “ground”, “wall” and “ceiling”. Through labeling, we can estimate the coarse geometric properties of an indoor scene, which can be used in a large number of applications, such as mobile robot navigation, object detection, automatic single-view or 3D reconstruction, virtual reality, video games, etc.

*Key–Words:* Geometric cues, Indoor image, Multi-class, Adaboost, Weak-learner, Decision tree, Labeling, Learning.

## 1 Introduction

Automatic 3D scene structure extraction from images is an active research area and can be used in a diverse set of applications, such as industrial manufacture, building and architecture, medicine and biometrics, multimedia, robotics and automation, etc. 3D information is very important for the applications of multimedia, especially in the applications of image-based rendering, virtual reality walkthrough, teleconference, distant education, video game design, etc.

Traditional methods use feature correspondence between pairs of stereo images or sequence of images, and/or projective geometry constraints to reconstruct the 3D constructions, which require special equipment, such as multiple cameras. Furthermore, using projective geometry constraints to recover a metric reconstruction of an architectural scene is computationally very intensive and usually unreliable.

Recently, Efros et.al [1] introduced a breakthrough method that allows to extract rough 3D structure from a single image. Unlike most of the scene recognition algorithms which model semantic classes, such as cars, faces, text, etc., [2, 3], they reconstructed the geometric structure of a scene through labeling (recognition). However, they focus their work on outside images, which are inherently different from indoor scenes. Our goal was to see if similar ideas can be used to construct a system for extracting 3D structure from indoor images.

This work is inspired by Efros’s work, and while there are overall similarities, we found that the structure has to be modified. In this paper, we use a large set of feature descriptions as geometric cues and apply a multi-class version of AdaBoost [4] with weak learners based on the decision tree to label an indoor image as “ground”, “wall” and “ceiling”. Our goal is to estimate the coarse geometric structure of an indoor scene through labeling.

## 2 Features and Training

Instead of acquiring all the required geometric parameters from a single image, in this paper, we use a large quantity of training images to obtain the required information through learning.

### 2.1 Features

Table 1 lists all the 35 features used in this paper, which include the information of location, color, geometry, texture and edges. The features are computed for image segments which can be obtained by image segmentation.

### 2.2 Training

In this paper, we use decision trees as weak learners and use AdaBoost to select a subset of discriminative features and boost weak learners. AdaBoost, a commonly used two-class (binary) boosting method,

Table 1: Features

| Feature Descriptions | No. |
|---|---|
| **Location** | 6 |
| L1. Centroid: x,y | 2 |
| L2. x,y: $10^{th}$ and $90^{th}$ percentile | 4 |
| **Geometry** | 4 |
| G1. Orientation | 1 |
| G2. Shape: ratio of MajorAxis/MinorAxis | 1 |
| G3. Eccentricity | 1 |
| G4. Area | 1 |
| **Edges** | 4 |
| E1. Compass Filters: mean | 4 |
| **Color** | 6 |
| C1. RGB values: mean | 3 |
| C2. HSV values: mean | 3 |
| **Texture** | 15 |
| T1. DOOG Filters: mean abs response | 12 |
| T2. DOOG Filters: mean of variables in T1 | 1 |
| T3. DOOG Filters: id of max of variables in T1 | 1 |
| T4. DOOG Filters: (max-median) of variables in T1 | 1 |

achieves a better overall regression/classification performance by combining a set of weak learners through an iteration procedure. It assigns different weights to training samples in such a way that different classifiers pay more attention to different samples. Since we have multiple labels, in this paper we apply the AdaBoost.M2 [5], a multi-class version of AdaBoost, which is briefly described as follows:

**Input:** sequence of $N$ examples $\langle(x_1,y_1),...,(x_N,y_N)\rangle$
with labels $y_i \in Y = \{1,...,k\}$
distribution $D$ over the examples

**Initialize:** the weight vector $w^1_{i,y} = D(i)/(k-1)$ for $i = 1,...,N, y \in Y - \{y_i\}$.

**Do for** $t = 1, 2, ..., T$

**1.** Set $W^t_i = \sum_{y \neq y_i} w^t_{i,y}$;
$q_t(i,y) = \frac{w^t_{i,y}}{W^t_i}, \quad D_t(i) = \frac{W^t_i}{\sum_{i=1}^N W^t_i}, \ y \neq y_i$

**2.** Call **WeakLearner**, with the distribution $D_t$;
get back a hypothesis $h_t : X \times Y \longrightarrow [0,1]$

**3.** Calculate the pseudo-loss of $h_t$:

$$\epsilon_t = \frac{1}{2}\sum_{i=1}^N D_t(i)(1-h_t(x_i,y_i)+\sum_{y \neq y_i} q_t(i,y)h_t(x_i,y))$$

**4.** Set the new weights vector to be

$$w^{t+1}_{i,y} = w^t_{i,y}\beta_t^{(1/2)(1+h_t(x_i,y_i)-(\frac{1}{k-1})h_t(x_i,y))}$$

for $i = 1,...,N, y \in Y - y_i, \quad \beta_t = \epsilon_t/(1-\epsilon_t)$

**Output** the hypothesis

$$h_f(x) = arg \max_{y \in Y} \sum_{t=1}^T (\log \frac{1}{\beta_t})h_t(x,y)$$

# 3 Implementation

In this paper, we first partition the training images into a large set of segments as shown in Fig. 1, where different colors are used to represent different segments. Then, we manually label those obtained segments as one of the three predefined labels, namely, “ground”, “wall” and “ceiling”. Labeled segments are then used as training samples and fitted in a boosted decision tree classifier. Instead of labeling all the obtained segments as training samples, in this paper, we only label those with fairly large areas. Fig. 2 illustrates the lablled training sample images and their corresponding ground truth, respectively, where the color red indicates “wall”, green indicates “ground”, and blue indicates “ceiling”.

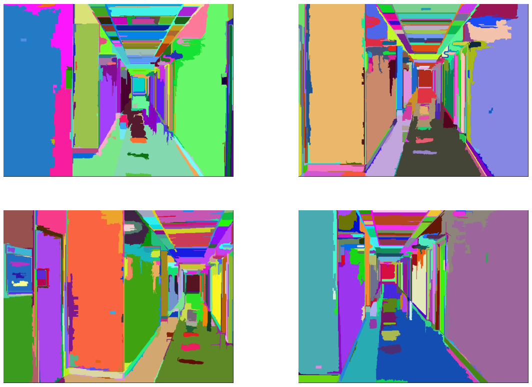
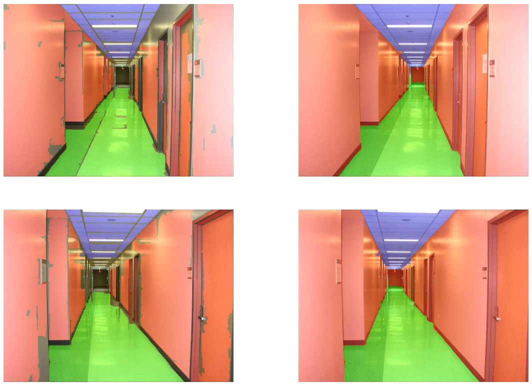


Figure 1: Over Segmented Images



(a) (b)

Figure 2: Labeled Images (a) labeled for training (b) ground truth

In this paper, the image partition is implemented using a publicly available code (C++), which is an implementation of graph-based segmentation method proposed by Felzenszwalb et.al [6].

The processing time for training a set of 1695 samples with 15 iterations is 4.5 hours and it costs around 32.0 seconds for testing a $480 \times 640$ image by using Matlab on a personal laptop with Intel Pentium(R) 1.8GHZ processor and 1.0G RAM.

# 4 Experimental Results and Discussion

In our database, we have 30 indoor images, where 1695 out of 6265 segments are manually labeled as one of the three predefined labels and used as training samples.

Fig. 3 $\sim$ 5 show the distributions of training samples with respect to different features. Fig. 3 shows that the location feature of centroid is a discriminative feature to distinguish "ground" and "ceiling". From Fig. 4 (a) we can see that orientation is also a good feature to distinguish "wall" from "ground" and "ceiling", as we can see that "wall" has a quite large distribution in a range around $90^o$ while the other two classes have large distributions in the range around $0^o$. Fig. 4 (b) shows that length ratio between major and minor axis is a also good feature to separate "ground" from the other two classes. Fig. 5 shows that neither the R-plane in RGB color spaces nor H-plane in HSV color spaces is a good feature.

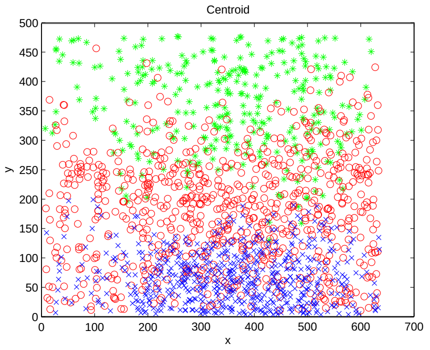
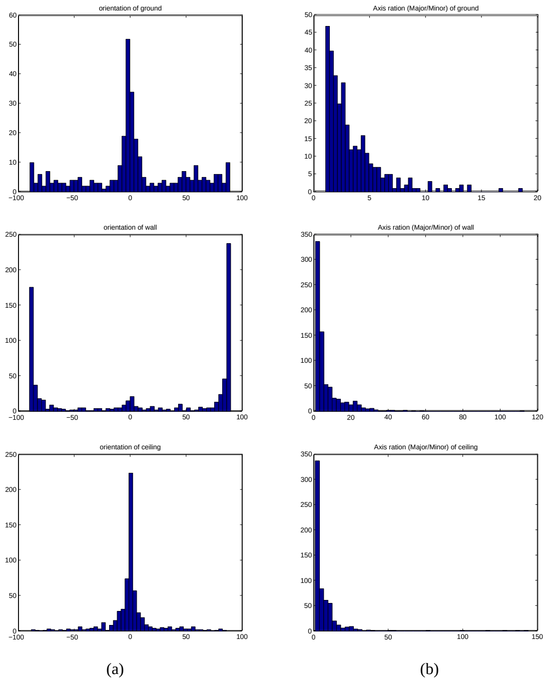


Figure 3: Training Sample Distribution (w.r.t location feature of centroid), where green $*$ represent ground samples, red $\circ$ represent wall samples, and blue$\times$ represent ceiling samples

## 4.1 Results

Fig. 6 and Fig. 7 show two samples of the experimental results, where Fig. 6 is a result of using only one single classifier and Fig. 6 is from the boosted classifier. From Fig. 6 we can see that different classifiers pay more attention to different samples, while Fig. 7 shows that boosting provides a better overall performance than every single classifier.

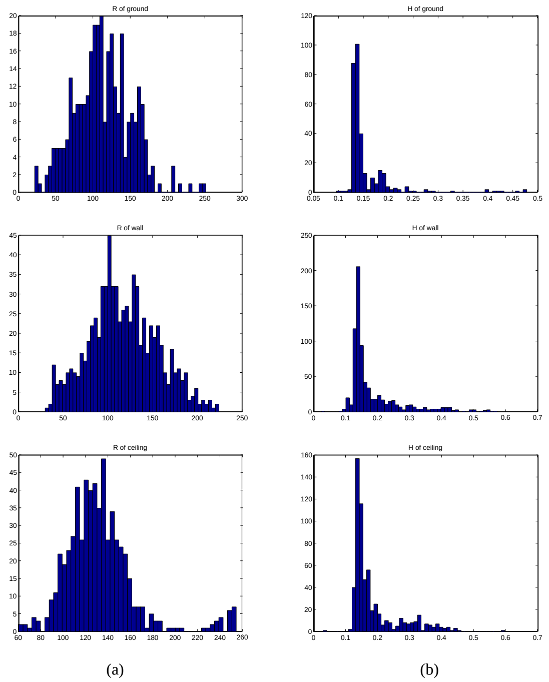
Table 2 shows the quantitative results, where we use a 5-fold cross validation to evaluate the perfor-



(a)

(b)

Figure 4: Training Sample Distribution (w.r.t geometry features) (a) histograms of orientation feature for ground, wall, and ceiling samples, respectively (b) histogram of ratio of Major axis length and Minor axis length for ground, wall, and ceiling samples, respectively



(a)

(b)

Figure 5: Training Sample Distribution (w.r.t color) (a) histograms of R plane for ground, wall, and ceiling samples, respectively (b) histogram of H plane for ground, wall, and ceiling samples, respectively

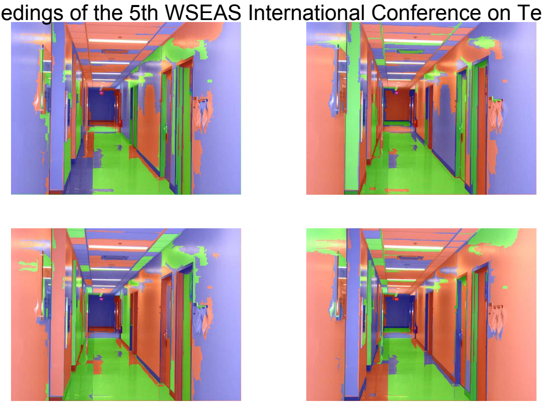


Figure 6: Results of Single Weak Classifier

Table 2: Confusion Matrix

| **Class** | Wall | Ground | Ceiling |
|---|---|---|---|
| Wall | 73.04% | 14.78% | 12.18% |
| Ground | 2.89% | 97.11% | 0% |
| Ceiling | 1.73% | 0.09 % | 98.17% |

mance and the confusion rate is calculated based on pixels.

From Table 2 we can see that the confusion rates between “ceiling” and “ground” are extremely low, while between “wall” and either “ceiling” or “ground” are a little bit high but reasonable. Thus, we can say that it can label “ceiling” and “ground” regions almost perfectly and can label “wall” reasonably well. This can also be seen from Fig. 7. The main reason lies on the composition of the feature set. For example, we can see location is a very good feature for distinguishing “ground” and “ceiling” as shown in Fig.3. In order to distinguish “wall” from the other two classes, more discriminative features are required.

### 4.2 Discussion

Since training samples are obtained through labeling image segments, the final results heavily depend on the performance of segmentation. There is a tradeoff in the number of segments in segmentation. Under segmentation increases the error rate, while over segmentation costs more computation time. In this paper, we aim to achieve high accuracy rate at the expense of computation time. On the other hand, the complexity of the decision tree and the composition of the feature set are also critical issues.

## 5 Conclusion

In this paper, we use a large set features such as information of colors, locations, geometry, edges and

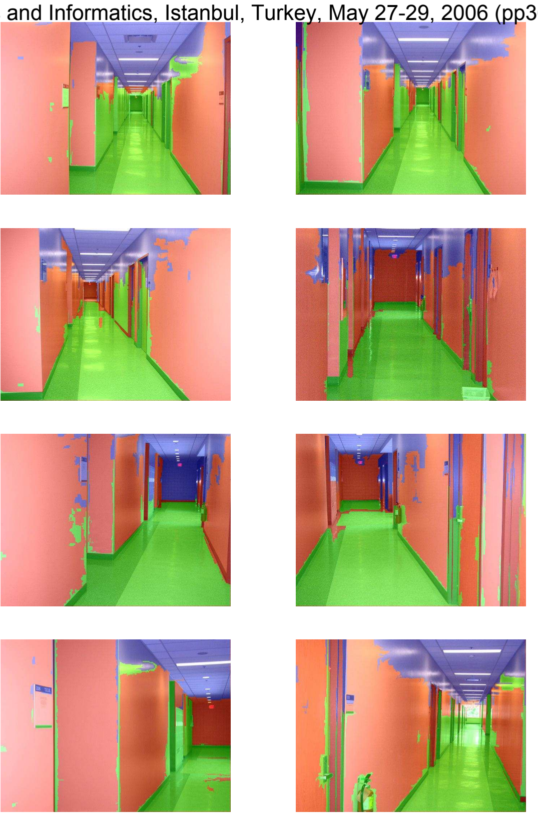


Figure 7: Labeled Images, where it uses AdaBoost to choose a subset of features (3 over 35) for every iteration ( iteration=15)

texture as geometric cues and apply a multi-class version of AdaBoost with weak learners based on decision trees to label an indoor image as “ground”, “wall” and “ceiling”. Through labeling, we can estimate the coarse geometric information of an indoor scene and it can be applied in many applications such as indoor robot navigation, object recognition, 3D reconstruction, virtual reality walkthrough, video game design, etc.

Our ultimate goal is to extract rough 3D scene structure from indoor scenes. Thus, the information of facing direction of wall, such as facing left, right, front, etc., is also necessary. As we mentioned above, the composition of feature set is also a critical issue. In conclusion, our main future work involves subclass labeling and adding more features into feature set.

**Acknowledgements:** Financial support for this research is provided by the Natural Sciences and Engineering Research Council (NSERC) and the University of Western Ontario.

*References:*

[1] D. Hoiem, A. Efros, and M. Hebert, "Geometric context from a single image," in *ICCV*, 2005, pp. –.

[2] A. Ferencz, E. Learned-Miller, and J. Malik, "Building a classification cascade for visual identification from one example," in *International Conference of Computer Vision (ICCV05)*, 2005.

[3] X. Liu and J. Samarabandu, "An edge-based text region extraction algorithm for indoor mobile robot navigation," in *Proc. of the IEEE International Conference on Mechatronics and Automation (ICMA 2005)*, Niagara Falls, Canada, July 2005, pp. 701–706.

[4] F. Y. and S. R. E., "Experiments with a new boosting algorithm," in *Machine Learning: Proceedings of the Thirteenth International Conference*, Morgan Kauffman, San Francisco, 1996, pp. 148–156.

[5] G. Eibl and K. P. Pfeiffer, "Analysis of the performance of adaboost.m2 for the simulated digit-recognition-example," in *Proceedings of the 12th European Conference on Machine Learning*, 2001, pp. 109–120.

[6] P.Felzenszwalb and D.Huttenlocher, "Efficient graph-based image segmentation," *IJCV*, vol. 59, no. 2, pp. –, 2004.