

# Consistent Line Clusters for Building Recognition in CBIR \*

Yi Li and Linda G. Shapiro

Department of Computer Science and Engineering

University of Washington

Seattle, WA 98195-2350

{shapiro,yi}@cs.washington.edu

## Abstract

*This paper introduces a new mid-level feature, the consistent line cluster, for use in content-based image retrieval. The color, orientation, and spatial features of line segments are exploited to group them into line clusters. The interrelationships among different clusters and the intrarelations within single clusters are used to recognize and roughly locate buildings in photographic images. Experiments are performed on a database of color images of outdoor scenes.*

*Keywords: object recognition, cluster features, CBIR.*

## 1. Introduction

With the rapid growth of storage devices, digital photographs, and internet access, content-based image retrieval (CBIR) has become increasingly popular in recent years. Systems that can automatically analyze, categorize, and search image databases have been developed both in research labs and by commercial concerns. Most of the early research systems perform retrieval based primarily on low-level image features, such as color and texture. They expect the user to provide a query image, which is an example of the images he wishes to retrieve. The query image features are extracted, and database images with similar features are returned, usually in order of similarity. The commercial versions of these systems have not proven to be very useful, because human users, who are looking for images for advertising, marketing, and other illustrations, think in terms of the high-level objects and concepts that should appear in an image, and they usually do not have an example image to show. Commercial systems, such as those provided by Corbis, Inc. and Getty Images, still use keyword indexing performed in advance by human technicians. They do not expect their users to possess or create a query image that has the exact characteristics they seek. Instead, these systems provide a rich vocabulary in which users describe the

objects and concepts to be found in the images they seek.

There is already a wealth of literature in the area of content-based image retrieval, much of it related to global color and texture features. Some efforts have been made to shorten the distance between human users and computer systems. One approach is to define *structural features* that attempt to capture the structure of a class of images [10] [11]. Another is to segment images into regions whose features are typical of certain well-known objects, such as tigers and zebras, which have characteristic color and texture [2] or characteristic shape [3]. Queries can request images that have regions with certain properties in certain spatial relationships [9]. Another approach [8] is to employ user relevance feedback to refine the query results, but this is usually paired with the query-by-example approach.

Our ultimate goal is to develop a system that can recognize a variety of objects and concepts in images and thus can be used for automated or semi-automated indexing of large image databases. Some examples of work on this level include people and horse recognition [6], building recognition [7], and high-level image classification systems [10]. We have developed new structural features called *consistent line clusters* that are useful in recognizing and locating man-made objects in images. Consistent line clusters are local rather than global image features. Thus they support both keyword indexing and spatial relationship queries. In this paper we will define the consistent-line-cluster feature, describe its implementation in terms of lower-level features, and discuss its use in building recognition.

## 2. Consistent Line Clusters

Lines are very important for object recognition, particularly for man-made objects. An important question for CBIR is how to use the extracted segments to form more advanced features that can be used to recognize various objects. Our solution is to group the lines into consistent line clusters and to use intra-cluster and inter-cluster relationships to recognize complex objects. We have applied this approach to building recognition and achieved promising results.

---

\*This research was supported by the National Science Foundation under Grants IRI-9711771 and IIS-0097329.

The lines from a typical building image are shown in Figure 1. Our algorithm uses the Canny edge detector [1] to find edges in the image and the Object Recognition Toolkit [4] to segment them into straight lines. One important characteristic of buildings is that they contain many line segments, often horizontal and vertical, which come from the boundaries of the windows, the doors, or the building itself. Another observation is that if two line segments come from different objects, the pixel colors around them are usually (though not always) different. In addition, line segments from different objects will cluster into different spatial groups. Based on these observations, we use the color, orientation, and spatial features of the line segments to partition them into consistent line clusters.

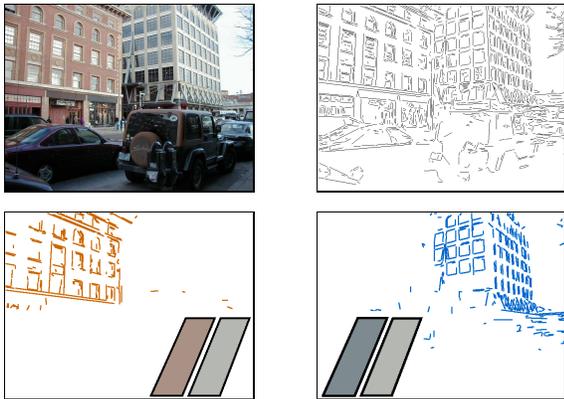


Figure 1: (top left) Original image. (top right) Line segments. (bottom) Color-consistent line clusters.

### Color-Consistent Line Clusters

Ideally, a line segment forms the boundary of two regions of different colors. Both colors are used for the color feature of the line, which we will refer to as a *color pair*. To reduce the complexity, we first classify each pixel of the image as one of several dominant colors, using the Gong color clustering algorithm [5] and the CIEL\*a\*b\* color space. Then each line segment is assigned one or more color pairs consisting of one dominant color from its left region and one from its right region, based on a small window of analysis. The line segments are grouped into color-consistent line clusters based on their color pairs. The main color pair of the left building in Figure 1 is (tan,gray), while the main color pair of the right building is (grayblue,gray). The two color clusters (bottom row) also contain spurious segments from other objects.

### Orientation-Consistent Line Clusters

The result of the color clustering is a set of line clusters, each of which corresponds to a specific color pair. For ev-

ery color-consistent line cluster, the orientation feature of the line segments can be used to further classify them. We would like to assign the parallel segments of an object to exactly one orientation-consistent line cluster. Because of the effect of perspective projection, the parallel lines on an object may not be parallel in the image, but will converge to a single point. Because of this, we use two steps to achieve our objective: first, roughly classify the segments according to their orientation in the image, and second, decide whether they are parallel to each other or they converge to a vanishing point in the image. Finding the roughly orientation-consistent line clusters is achieved through a simple clustering algorithm that finds the peaks in the orientation histogram and assigns each line segment to the cluster associated with its closest peak. After the roughly-orientation-consistent line clusters are obtained, the perspective information is used as a key both to decide whether the segments in a line cluster are consistent and to filter out the “noise” lines. Each of the two color clusters in Figure 1 produced several orientation-consistent clusters as shown in Figure 2.

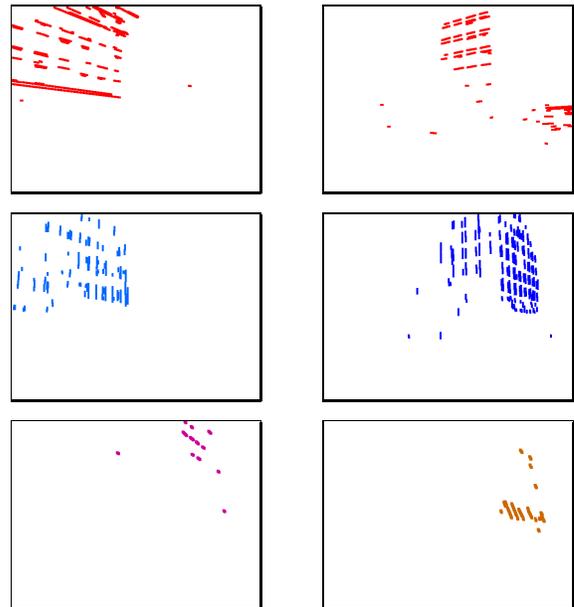


Figure 2: Orientation-consistent line clusters obtained from the color-consistent line clusters shown in Figure 1. The results are final orientation-consistent clusters using both orientation and perspective information with small clusters removed.

### Spatially-Consistent Line Clusters

After constructing the consistent line clusters using color and orientation features, the resultant clusters may still have some segments from different physical entities. To rule out such segments, spatial clustering is performed using

both vertical and horizontal position histograms. First, the line segments in a cluster are projected to the y-axis to create a vertical position histogram, which can be segmented into groups of y-positions that yield vertical position clusters. Then, the line segments of each vertical position cluster are projected to the x-axis to create a horizontal position histogram whose segmentation produces horizontal position clusters. The line segments in the resultant spatially-consistent line clusters are close to each other, both vertically and horizontally, in the image. The application of color-consistent clustering followed by orientation-consistent clustering followed by spatially-consistent clustering yields a set of *consistent line clusters* that are used to detect buildings or other line-segment-rich structures. Figure 3 shows two spatially-consistent line clusters which came from the single orientation-consistent line cluster in the top-right position of Figure 2. The cluster has been divided into the line segments from a building and those from an automobile.

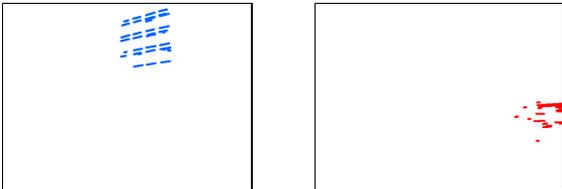


Figure 3: Two spatially-consistent line clusters obtained from the single orientation-consistent line cluster shown in Figure 2 (top-right image).

### 3. Building Recognition

Once the consistent line clusters have all been constructed, they can be used to detect objects, such as buildings. We use two criteria to detect buildings: the interrelationships of the consistent line clusters detect structure-preserving or *junction-rich buildings*, and their intra-relationships can be used to detect the simple-structured or *overlap-rich buildings*. These two situations are shown in Figure 4 and discussed below. The location of a building can be estimated from the position of its corresponding line clusters.

#### Interrelationship Criterion

Because many line segments on buildings are from windows and doors, there will be many intersections. Two line segments are classified as intersecting if when they are extended, the intercept point of the two virtual lines is on one of the line segments or is close to one of their end points. For every line cluster pair  $(cl_i, cl_j)$ , the interrelationship between the two line clusters is considered to decide whether this cluster pair is a hint of the existence of buildings. In

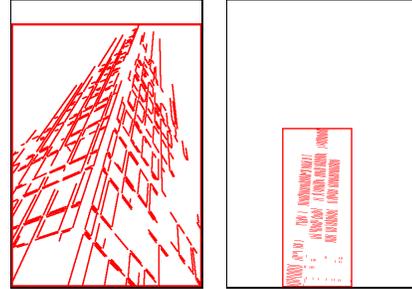


Figure 4: (left) Interrelationship criterion. (right) Intra-relationship criterion.

order for the pair to be a qualified hint, its line segments should form enough intersections. This can be represented by  $N_{inter_{ij}}$ , the total number of lines in  $cl_i$  that intersect some line in  $cl_j$ . For each cluster, we can then define  $N_{inter_i}$ , the maximum number of intersecting lines of  $cl_i$  over all other clusters, and finally, its normalized version  $Norm\_inter_i$ , normalized by the total number of lines in the cluster.

$$N_{inter_{ij}} = |\{l_1 \in cl_i \mid \exists l_2 \in cl_j, l_1 \text{ intersects } l_2\}|$$

$$N_{inter_i} = \max_j \{N_{inter_{ij}} \mid i \neq j\}$$

$$Norm\_inter_i = \frac{N_{inter_i}}{|cl_i|}$$

#### Intra-relationship Criterion

Due to the many different appearances of buildings and the different distances at which the images were taken, some buildings do not produce many junctions on the image, but they do have many overlapping line segments. The intra-relationship criterion is used to examine how many lines heavily overlap in a line cluster  $cl$ . For a line  $l$ ,  $Nol(l) = |\{l' \mid l' \text{ overlaps } l\}|$  is the number of lines in  $cl$  that overlap with  $l$ . If  $Nol(l)$  is large enough (greater than a learned threshold  $T_{ho}$ ), then  $l$  is a heavily overlapped line. The number of heavily overlapped lines is another hint of the existence of buildings. Similar to the interrelationship criterion, the intra-relationship criterion is defined by features:  $N_{intra_i}$ , the number of heavily overlapped lines in the line cluster  $cl_i$ , and  $Norm\_intra_i$ , normalized by the total number of lines in the cluster.

$$N_{intra_i} = |\{l \mid l \in cl_i \text{ and } Nol(l) \geq T_{ho}\}|$$

$$Norm\_intra_i = \frac{N_{intra_i}}{|cl_i|}$$

### 4. Experimental Evaluation

Our consistent-line-cluster (CLC) features can be used for two different tasks: 1) content-based image retrieval and 2)



buildings buildings cactus

Figure 5: Sample images from the online test set.

object recognition. For content-based image retrieval, we built a simple decision-tree classifier. The feature vectors for this task were designed to convert the local features for the separate clusters to a global histogram as follows:

$$Ninter(k) = \max\{Norm\_inter_i | Ninter_i \geq k\}$$

$$Nintra(k) = \max\{Norm\_intra_i | Nintra_i \geq k\}$$

for  $k = 1$  to  $max_k$ . With the feature vectors

$$(Ninter(1), \dots, Ninter(max_k), \\ Nintra(1), \dots, Nintra(max_k))$$

and  $max_k$  set to 64, we used the C4.5 package to generate simple decision-tree classifiers. Our test database of 977 images was obtained from two online image databases: creatas.com and freefoto.com. We selected 336 building images and 641 nonbuilding images and ran a set of cross-validation experiments in each of which 90% of the images were used as the training set and the other 10% as the test set. The average error (false positives plus false negatives) over the set of cross-validation experiments was 5.8%. (Note that our algorithms were developed on a completely independent database of our own images of campus, city, and landscape scenes.)

There are three CBIR methods that seem most related to our own: Iqbal and Aggarwal’s approach [7] to building recognition using perceptual grouping (rectangles), Zhou, Rui, and Huang’s water-filling algorithm [11] for extracting edge-map features, and Vailaya, Jain, and Zhang’s edge-direction-histogram (EDH) features [10] for classifying city vs. landscape images. Because the emphasis in [7] and [11] was quite different from our own and since several early reviewers of our paper suggested comparing our features to the EDH features, we implemented EDH and tested it on classifying building vs. nonbuilding images, using C4.5 and cross validation as above. On the same test set, the EDH method had an average error of 16.5% compared to CLC’s 5.8%. Note that this test data set (see Figure 5 for examples) is a fairly difficult one with the building images

taken from many different viewpoints and the nonbuilding images containing many vertical line segments. We found that EDH worked very well on images with many vertical lines and less well on those without vertical lines. Our CLC features are not sensitive to a particular orientation and can also handle perspective.

The CLC features can also be used for object recognition and approximate localization, so that the presence and locations of objects in images can be detected and indexed for future retrievals. For object recognition, we tested the algorithm on 97 well-patterned buildings, 44 non-well-patterned buildings, 16 non-patterned non-buildings, and 25 patterned non-buildings in a local test set, specifically acquired to control these experiments. The results were 0% error for well-patterned buildings, 4.5% error for non-well-patterned buildings, 6.2% error for non-patterned non-buildings, and 100% error for the patterned non-buildings, which were objects like faculty mailboxes, buses, and fences, all selected to try to fool the algorithm (and they all did!). Additional features (context) will be required to distinguish these similarly structured objects from buildings. Figure 6 shows some correct classification/location results on this image set. The first and second row images were considered to contain well-patterned buildings, while the third and fourth row images, which are more difficult, were considered to be non-well-patterned.

Two of the misclassifications are shown in Figure 7. The reasons for false negatives are the lack of enough patterns and the pattern interference from other objects, for example, trees. Some patterns from objects other than buildings, such as trees and bridges, are recognized as buildings in the false positives.

Two images from the patterned non-buildings are shown in Figure 8. Although they are currently false positives, they also show the potential use for consistent line clusters, along with other features, for recognizing additional man-made objects.

## 5. Summary

We have described an algorithm that uses the low-level features of extracted line segments to assign them to consistent line clusters, new mid-level features that can be used for high-level object detection and location. The experiments on buildings show that it provides good results both in image retrieval and in object recognition. Our algorithm was able to find both close and far-away buildings from many different viewpoints and in many different contexts. It does not rely on vertical line segments or rectangular window features. Furthermore, it outperformed the EDH features of

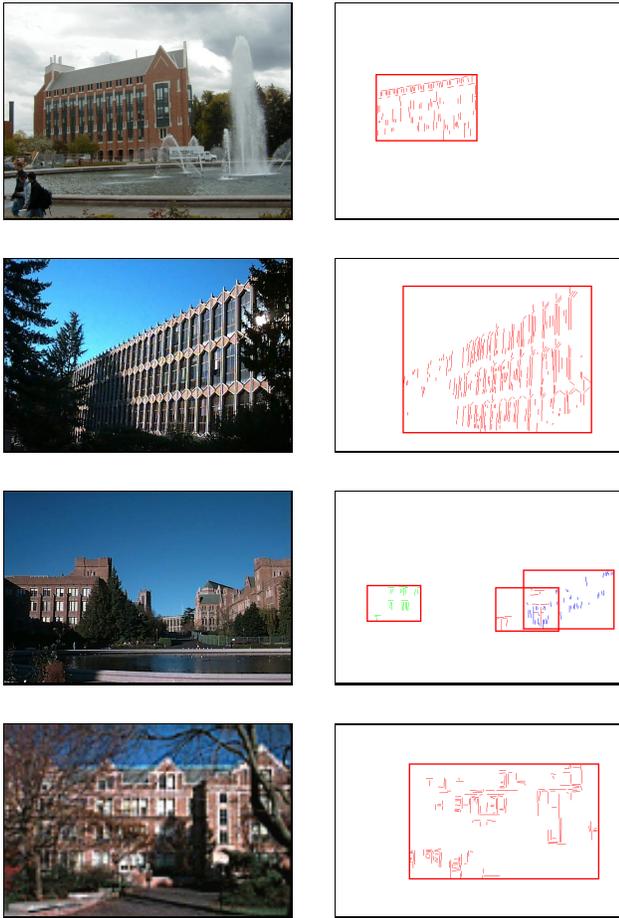


Figure 6: Some correct classifications.

[10] in classifying images of an independent web-derived dataset as building or nonbuilding. We intend to use consistent line clusters as part of a large set of both low- and mid-level features in a system that will learn to recognize many different common objects found in photographic images of outdoor scenes.

## References

- [1] J. Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 8, pp. 679-698, 1986.
- [2] C. Carson, S. Belongie, H. Greenspan, J. Malik, "Region-based Image Querying," *Proceedings of the 1997 IEEE Workshop on Content-Based Access of Image and Video Libraries*, pp. 42-49, June 1997.
- [3] A. Del Bimbo, P. Pala, S. Santini, "Visual image retrieval by elastic deformation of object sketches," *IEEE Symposium on Visual Languages*, pp. 216-223, 1994.



Figure 7: False negative (left) and false positive (right).



Figure 8: Two false positives that show the potential for using consistent line clusters to find additional man-made objects.

- [4] A. Etamadi, "Robust Segmentation of Edge Data," *Proceedings of the IEE Image Processing Conference*, 1992.
- [5] Y. Gong, "Advancing Content-Based Image Retrieval by Exploiting Image Color and Region Features," *Multimedia Systems*, Vol. 7, pp. 449-457, 1999.
- [6] D. Forsyth and M. Fleck, "Body Plans," *IEEE Society Conference on Computer Vision and Pattern Recognition*, pp. 678-683, 1997.
- [7] Q. Iqbal and J. K. Aggarwal, "Applying Perceptual Grouping to Content-Based Image Retrieval: Building Images," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 42-48, June 23-25 1999.
- [8] Y. Rui, T. Huang, and S. Mehrotra, "Relevance Feedback: A Power Tool for Interactive Content-Based Image Retrieval," *Proceedings of the IEEE Workshop on Content-based Access of Image and Video Libraries*, pp. 67-74, 1997.
- [9] J.R. Smith and S.-F. Chang, "VisualSEEK: A Fully Automated Content-Based Image Query System," *ACM Multimedia*, pp. 87-98, November, 1996.
- [10] A. Vailaya, A. K. Jain and H.-J. Zhang, "On Image Classification: City Images vs. Landscapes", *Pattern Recognition*, Vol. 31, pp 1921-1936, 1998.
- [11] S. X. Zhou, Y. Rui, and T. S. Huang, "Water-filling Algorithm: A Novel Way for Image Feature Extraction Based on Edge Maps," in *Proc. IEEE Int. Conf. on Image Proc.*, 1999.