1998 International Computer Symposium
Workshop on Image Processing and Character Recognition
December 17-19, 1998, N.C.K.U., Tainan, Taiwan, R.O.C.

Line Detection from Photo Images and Image Retrieval using Line Information

Seiichi KON'YA *Hiroki AKAMA* *Masashi YAMAMURO*

NTT Information and Communication Systems Laboratories.
1-1 Hikarino-oka, Yokosuka-Shi, Kanagawa 239-0847, Japan
E-mail:{konya, akama, masashi}@dq.isl.ntt.co.jp

Abstract

In almost all photo images, one can see various straight lines such as horizontal lines, the outlines of tree trunks, buildings, door frames and so on. This line information can be used to characterize these images as well as the objects in the images. This paper proposes a new line detection algorithm based on the Hough transform. We improve the efficiency of parameter estimation by using edge direction information and reduce the computation time with an edge thinning method. Experimental results on 1010 PhotoDisc images show that 40 percent of detected lines can be utilized for image retrieval. We study the image similarity retrieval which utilizes the detected lines as features of images. In order to retrieve images of various aspect ratio, we use a new method that normalizes the offset of the lines. We implement a simple image retrieval prototype and show that the normalization method works effectively in retrieving the images containing lines similar to the reference specified by a user.

keywords: line detection, offset normalization, Hough transform, wavelet transform, image retrieval, similarity search, image database

1 *Introduction*

We are developing an image database system ExSight, which automatically extracts objects from images and provides feature-based image retrieval [1, 2]. ExSight provides users with a content-based retrieval that searches images containing objects similar to the reference specified by a user from the viewpoint of color, shape, size and position. Adding the impression which a user receives from the image to the feature attributes makes image retrieval more powerful. In photo images, for example, there is various straight-line information such as horizons, trees, buildings, door frames and so on. This line information can characterize the image as well as the objects in image.

This paper proposes a line detection algorithm and uses it for image retrieval in an experiment involving 1010 PHOTODISC [3] images.

2 *Hough Transform*

The Hough transform [4] is a popular method for detecting line and curve segments in an image. Typically, the straight line l is parameterized by the Hessian normal form (ρ, θ), where θ is the angle of the normal to the line and ρ is the distance form the origin of the image (Fig. 1). Using this parameterization, all points (x, y) on the line l is will satisfy the equation:

$$\rho = x \cdot \cos\theta + y \cdot \sin\theta, \tag{1}$$

There are infinitely many lines which pass through the given point p (Fig. 2), however, the parameters of the line group passing through p satisfy the equation (1) and are shown as the solid curve in Fig. 3. The parameters of the line group passing through q are the dotted curve in Fig. 3, and the two curves intersect in Hough space at the point corresponding to the line passing through both p and q.

The lines in an image can be detected by computing the edge of the image and calculating the parameter sets of each edge pixel. Next, the Hough space is discretized into grid cells and each edge pixel "votes" on the cells for which its parameters pass through.

3 *A New Line Detection Algorithm*

The standard Hough transform, needs to calculate the parameter sets satisfying equation (1) for each edge pixel. In addition, a two-dimensional histogram is needed to record the "votes." Moreover, the histogram becomes complicated when the image contains many lines, with multiple maxima, many of which do not correspond to lines in the image but instead, are artifacts of noise.

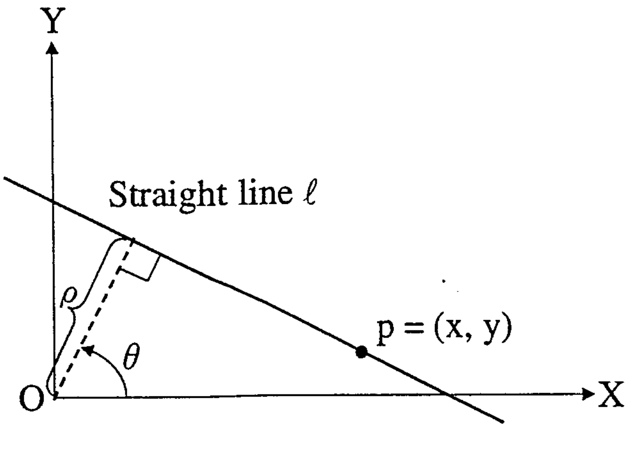

Figure 1: The Hessian normal form.

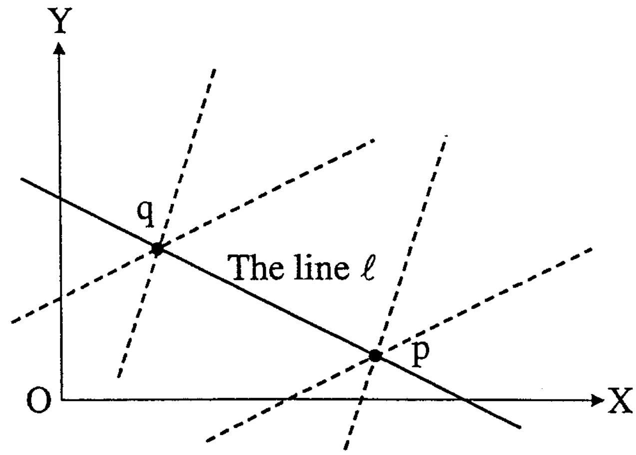

Figure 2: The coordinates of pixels and lines.

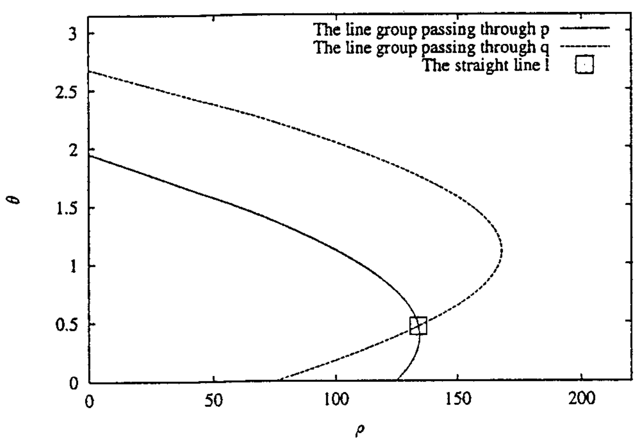

Figure 3: The Hough space.

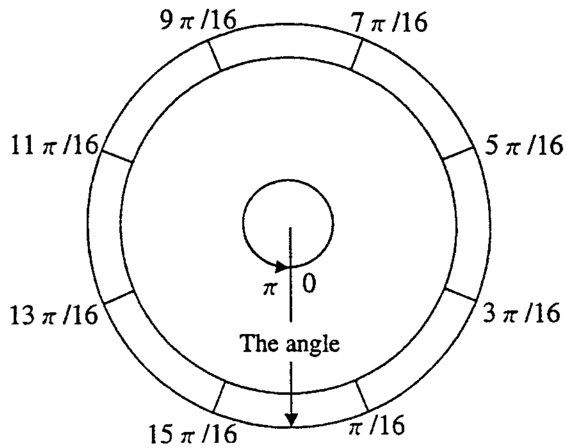

Figure 4: Example — discretizing in 8 stages.

Several variants of the Hough transform exist. We propose a line detection algorithm that uses an improved Hough transform that includes edge direction.

The proposed algorithm consists of the following steps: detecting edge with the wavelet transform, combining edges of different color, thinning of the edge map, estimating the line candidates, merging the line candidates, calculating the line segments.

3.1 Edge Detection using the Wavelet Transform

There are several edge detection algorithms which compute both the edge magnitude and the edge direction of images [5, 6]. In particular, the Mallat's wavelet transform [6] has multiscale properties that can deal with the contours of small structures as well as the boundaries of larger objects. A fine scale is best for detecting sharp lines. A coarse scale, on the other hand, can detect rough lines.

3.2 Combining the Color Edge Components

Edges are detected independently in each of the three color components such as RGB or CIE $L^*a^*b^*$. The edges of color components are combined using the method developed by H.-C. Lee et al. [7]. This method reduces the noise using the fact that signals of different color components are more correlated than the noise contained in the color components.

3.3 Thinning Edge Map

The edge map, calculated by the previous section, can be regarded as "mountain chains." The height of a mountain is its edge magnitude. If the edges are spread out, the accuracy of the estimated line parameters is low, and nearly parallel lines cannot be distinguished. Moreover, as the number of edge pixels increases, the parameter estimation computation time also increases. Thus it is necessary to reduce the number of unnecessary edges. Edge reduction method that try to set thresholds for the edge magnitude have difficulty in setting the threshold value and in discerning nearly parallel lines. Instead, we trace "the mountain ridge" by selecting the local maxima in the edge direction. This *skeleton* representation decreases the computation cost without neglecting weak lines and improves discernibility of nearly parallel lines.

3.4 The Parameter Estimation of Lines

The normal orientations are uniquely determined from the edge directions. The calculation of the parameter (ρ, θ) is once per edge pixel. The Hough space becomes sparse, consequently, the "votes" can be recorded on a tree structure instead of the 2-d histogram.

First, the Hough space is discretized into grid cells. The parameter of each edge pixel is calculated and the "vote" is recorded in the cell whose range includes the parameter. The cell whose degree is more than a certain threshold is a candidate for a line. This threshold corresponds to minimum length of the line to be detected.

3.5 Merging the Line Candidates

If the discretization step of the Hough space is too small or the parameters of the line are near the boundary of cells, the edge pixels that compose one line are split into multiple cells. To detect the line accurately, it needs to merge the line candidates split into multiple cells.

As the criterion to merge 2 cells, we employ the residual of the edges in one cell with respect to the line composed by edges in other cell. The residual$_{st}$ of cell t with respect to the line of cell s is:

$$\frac{1}{N_t}\sum_{i=1}^{N_t}\{e_y^s(x_i^t - \mu_x^s) - e_x^s(y_i^t - \mu_y^s)\}^2, \qquad (2)$$

where (x_j^t, y_j^t) and N_t are the coordinates and the number of edge pixels in cell t, respectively. μ^s and e^s are the center of gravity and the tangent vector of the least-squares approximation of cell s, respectively.

The two cells whose residual is less than a certain threshold are merged to one.

3.6 Calculating Line Segments Information

Straight lines in most images usually have the starting and ending points. To determine which segment (interval) of the line appears in the image, the edge distribution along the line is computed and the high density intervals are regarded as line segments. Line segment with lengths less than 20 pixels are removed as noise.

4 The Image Retrieval using the Line Information

4.1 Indexing the orientation of lines

The angle of line orientation θ is in the range $0 \leq \theta \leq \pi$. The similarity measure between the neighborhood of 0 and the neighborhood of π must be high. Two different methods are used to index of the orientation of the lines. The procedures are as follows.

- Discretize the angle so as to contain 0 and π in a same neighborhood(Fig.4):
- In order to utilize tree index such as B-tree, make the redundant entry of the line l according to their tangent θ:

duplicate the entry at $\theta + \pi$	$\theta < \frac{\pi}{4}$
duplicate the entry at $\theta - \pi$	$\theta > \frac{3\pi}{4}$
do not duplicate	otherwise

This method provides a range query within $\tau \pm \Delta\tau$, where τ is the given query key. Moreover, we can form the tangent, the offset and the length of lines into a 3-dimensional vector and perform a nearest-neighbor search with multi-dimensional index such as R-tree.

4.2 Normalizing the Offset of Lines

The offset of the line depends on the image size. If we want to retrieve images of different size in the same way (e.g., "search the images containing the line located at relatively the same position."), a normalization method that takes the image aspect ratio into account is required. It is desirable that all images in a database are normalized to same size, however, the change in aspect ratio of image makes the orientation of line change as well. Consequently, the search results may differ from the users' point of view. If images are centered with the extra borders, it works well to search the lines near the center. It does not work well to search "the line near the border" such as a line near the right or left border of a portrait image and the line near the top or bottom border of a landscape image. If, however, we clip images into the squares, the lines that have been clipped out can not be retrieved.

We introduced a normalization method that takes both the orientation of the line and the aspect ratio into account. In a portrait image, if a line is lateral, i.e., the angle of tangent θ is in the range $\theta \leq \frac{\pi}{4}$ or $\frac{3\pi}{4} \leq \theta$, the offset is not changed. If a line is longitudinal, i.e., $\frac{\pi}{4} < \theta < \frac{3\pi}{4}$, the offset is adjusted to enlarge the distance from the origin as the line nears the vertical (Fig. 5). In the case of a landscape image, we switch the vertical and the horizontal and do the same adjustment. In this way, we can search "the line near the border" regardless of the aspect ratio of the images.

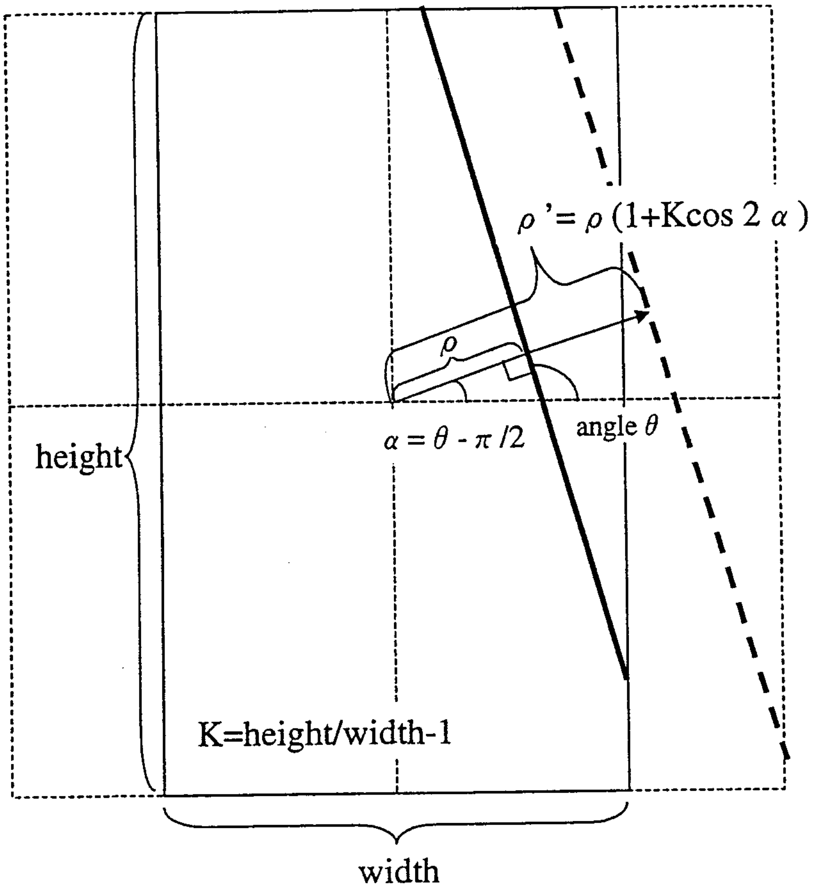

Figure 5: The offset normalization of lines.

4.3 *The Prototype System*

Human perception is sensitive to the orientation of lines. The offset of the line is also an important key. We implemented a prototype which filters images using the orientation and the offset of the line and ranks the images according to the length of the line contained in them.

5 *Experiments*

5.1 *Line Detection*

Figure 6 shows a sample color image (PHOTODISC Object Series 04019 [3]). Figure 7 shows the line segments detected by the proposed algorithm. There was a little displacement of the line from the original, however, the proposed algorithm found almost all lines correctly. The computation time, which varies depending on the image size and the number of lines, was about 25 seconds in this example when the algorithm was run on a DEC AlphaStation 250.

We applied the proposed algorithm to 1010 images of PHOTODISC and evaluated the ratio of relevant lines to detected lines. "The relevant lines" were the lines which are not artifacts of noise, skewed lines due to over-merging or line fragments due to over-splitting. They were then made available as the image features which were used to provide the similarity search.

The experimental results are summarized in Table 1. The rate of relevant line detection was about 40% because of short lines owing to the artifacts of noise or over-splitting at the calculating the line segment step. Improving the relevance rates requires a modification of the interval estimation step and the removal of isolated short lines. The relevance rates of "Food_Essentials" and "Food_and_Dining" are low because of the many curves in such objects as fruits and plates.

5.2 *The Image Retrieval*

Figure 8 displays the GUI which input the query key of the image retrieval prototype. The white area shows the query range and the thin line was the center line. The query key (plotted with thick line) is specified by the orientation and a point on the line. Figure 8 shows the case where the retrieved images contain a line near the right border.

Table 1: The evaluation of lines detected using the proposed algorithm.

Contents name (the num. of images used)	the avg. num. of lines	the relevant lines (percentage)
Children_of_the_World (69)	17.0	7.2 (42%)
European_Landmarks_and_Travel (69)	21.6	9.5 (43%)
Family_and_Lifestyles (160)	16.7	8.3 (50%)
Food_Essentials (160)	39.6	9.5 (24%)
Food_and_Dining (69)	23.6	6.4 (27%)
Homes_and_Gardens (69)	33.2	18.6 (56%)
International_Sports (336)	15.4	6.8 (43%)
People_Lifestyles_and_Vacations (69)	17.5	8.1 (46%)
The_Painted_Table (9)	19.6	9.7 (49%)
	21.9	8.6 (39%)

The prototype filtered the image with the angle and the offset information and displayed the top 10 image ranked by the line length (Fig. 9). Because of the normalization method, the prototype was able to retrieve images containing the vertical line near the right border regardless of their aspect ratio.

6 Conclusions

This paper proposed a new line detection algorithm for similarity search of color images which retrieves the image containing the line similar to a reference specified by a user. The proposed line detection algorithm consists of the following steps: 1) detecting edges using the wavelet transform, 2) combining color edges, 3) thinning edge map, 4) estimating the line parameters, 5) merging the line candidates, 6) calculating the line segments. The proposed algorithm was applied to PhotoDisc 1010 images and 40% of detected lines were relevant to the similarity search.

We also studied content-based image retrieval utilizing the detected lines as image features. The line offset normalization method was introduced to the similarity search on databases containing images of various aspect ratios. A simple prototype showed that our normalization method worked effectively.

References

[1] H. Akama, K. Mii, S. Kon'ya, K. Kushima, "ExSight — Image Retrieval System based on Automatic Object Extraction," 8th Workshop of IEICE Data Engineering (DEWS '97), pp. 107-112.

[2] H. Akama, S. Kon'ya, K. Mii, K. Kushima, "ExSight — Image Retrieval System based on Automatic Object Extraction applied to Photographs (PhotoDisc)," Technical report of IEICE, DE97-31, 1997-07.

[3] http://www.photodisc.com/

[4] R. Klette and P. Zamperoni, "Handbook of Image Processing Operators," John Wiley & Sons Ltd., 1996, pp. 350-355.

[5] J. Canny, "A Computational Approach to Edge Detection," IEEE trans. on PAMI, Vol. 8, No. 6, Nov. 1986.

[6] S. Mallat and S. Zhong, "Characterization of Signal from Multiscale Edges," IEEE trans. on PAMI, vol. 14, No. 7, July 1992.

[7] H.-C. Lee and D. R. Cok, "Detecting Boundaries in a Vector Field," IEEE trans. on SP, vol. 39, No. 5, May 1991.

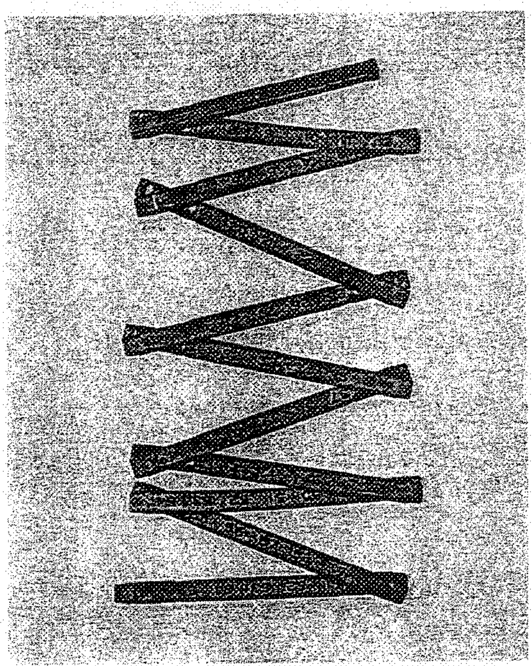

Figure 6: The original image

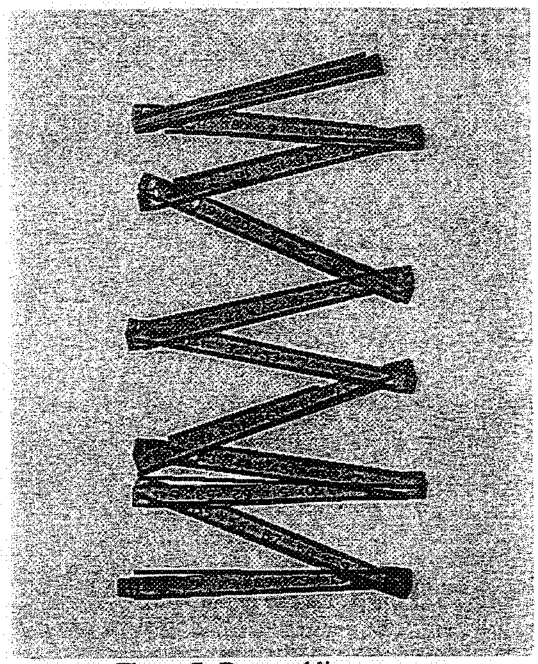

Figure 7: Detected line segments

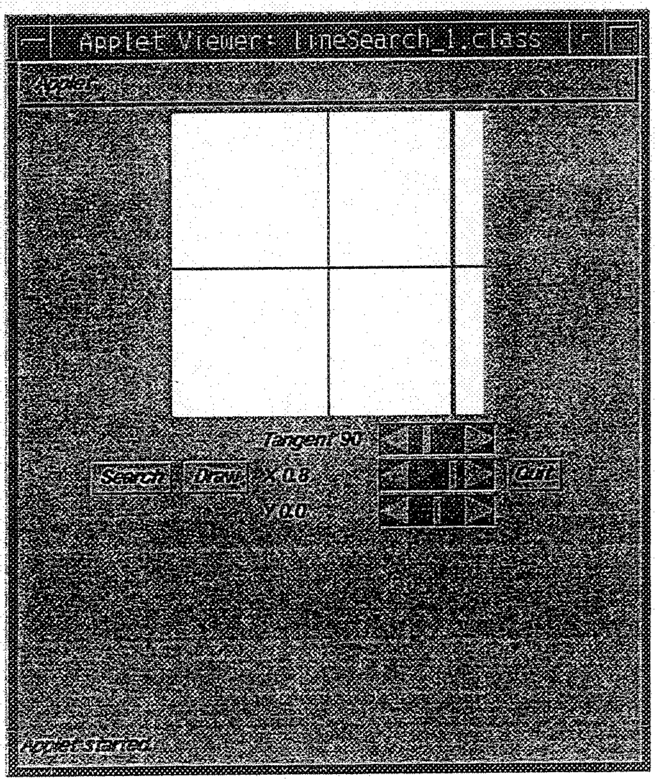

Figure 8: query key

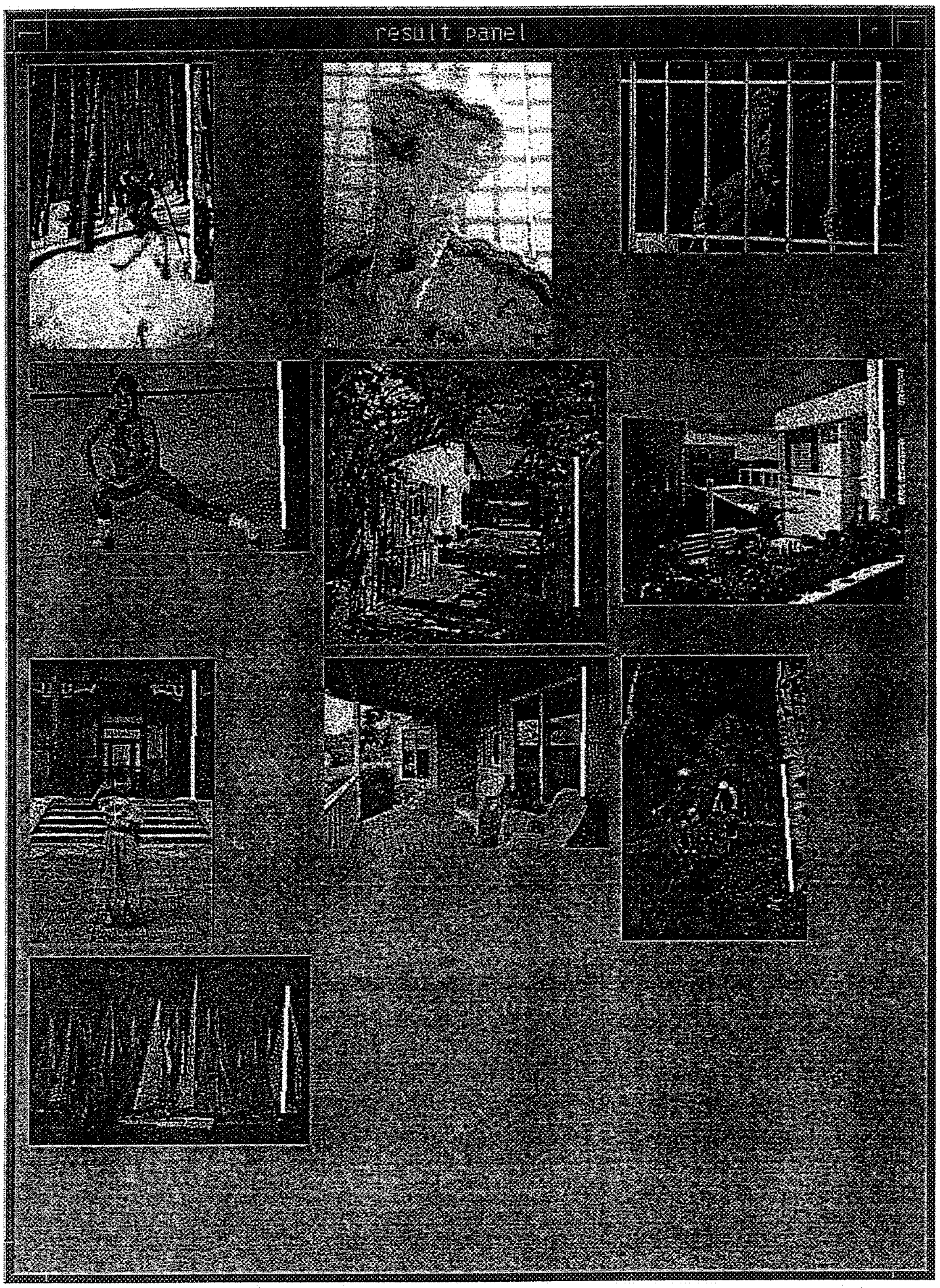

Figure 9: The search results