A Comparison of Line Detectors for Image Background Modelling

Sébastien Lefèvre, Cédric Jeusse, and Nicole Vincent

Laboratoire d'Informatique, Université de Tours
64 avenue Portalis, 37200 Tours – FRANCE
{lefevre,vincent}@univ-tours.fr

Abstract. In the context of multimedia data analysis, it is often necessary to understand what happens in a scene, which can rely on detecting and tracking the different moving objects and by comparing their positions at a given time with their environment. In order to model the environment or the scene background, we consider straight lines as appropriate features. In this paper, we compare some classical line detection approaches with two low cost contributed techniques, particularly relevant for real time scene analysis. The comparison methodology involves the definition of some evalutation measures. Results obtained show that a contributed method, called the connection approach, gives the best results.
Keywords: line detection, background modelling, scene understanding.

1 Introduction

Multimedia information systems contain huge amounts of information. Appropriate and efficient tools are needed to correctly structure and understand the multimedia information. When dealing with image sequences, it corresponds to scene understanding, which can rely on the analysis of the different objects present in the scene and their respective positions at a given time. However, object tracking gives only these positions in the 2-D image space, whereas it is necessary to know these positions relatively to a more global context, *i.e.* the scene background. In this paper, we are concerned with techniques for extracting and modelling the scene background.

In a first section, we will justify why lines can be considered as the most appropriate primitives for background modelling. Then we will briefly recall some classical line detection approaches and introduce two original ones. Finally we will compare the different approaches in terms of efficiency and quality.

2 Lines as appropriate features to model the image background

In order to describe a scene with a minimum set of details, it is possible to rely on a drawing, which consists mainly of strokes and rarely of filled areas. Moreover, main lines of the environment to be analysed are most of the time straight lines. Beyond this conceptual principle, we have to verify the significance of straight lines as image primitives for background modelling when considering a computational point of view.

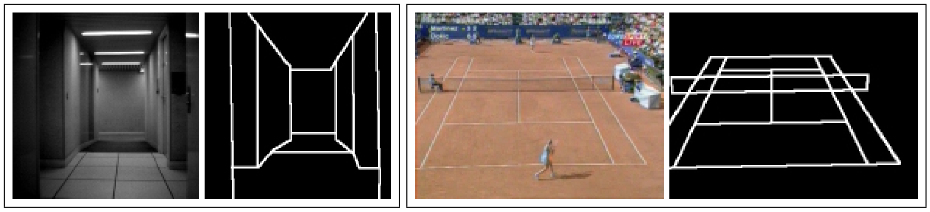


Fig. 1. Background modelling of an indoor scene (left frame) and an outdoor scene (right frame) using straight lines: the original images (left) and the background models (right).

Among classical features which can be computed on images, two categories can be considered. In the one hand, we can focus on homogeneous parts of the image. Features used are then relative to regions (for example some texture attributes) but have two main disadvantages from a computational point of view: the lack of precision and the processing time required. On the other hand, we can focus on heterogeneous parts of the image. The features represent here the discontinuities which appear in the image: corner points, edges, or lines. Corner points can be extracted efficiently, but they are very sensible to occlusion phenomena. On the contrary, edges are less sensible to occlusion due to the amount of edge points which can be detected, but this important quantity of information brings the problem of the complexity of the background model.

Finally, straight lines are characterized by a relative robustness to occlusion phenomena and can be extracted in real time (see section 5). Figure 1 shows two examples where lines are appropriate features for background modelling.

The choice of straight lines has been justified from both a computational and a human point of view. We are now to review some classical approaches for line detection.

3 Classical line detectors

The goal of this section is not to present an exhaustive panel of the literature, but rather to focus on the approaches used in our comparison framework. These methods are either Hough Transform-based methods, or local methods.

The Hough Transform is surely one of the most well-known approaches. This technique can be applied on binary edge images in order to detect lines considering a parametrical representation. The method relies on the use of an accumulator array where each cell represents a given physical line (modelled in the polar representation space using ρ and θ). For every edge point, all the lines (ρ, θ) the point can belong to are considered and the associated values are incremented. The highest values in the array represent most plebiscted lines and are extracted. Despite its robustness to image noise, this method is considered to be greedy in memory and computational resources. So several methods have been derivated from it. The Probabilistic Hough Transform [4] relies on a sampled edge image, obtained from a random selection of edge points. Hough Transform is then applied on this subset, giving a lower computation time but also a lower results precision. The Combinatorial Hough Transform [1], computes for every two edge points the properties of the associated line and increment the relevant accumulator value. The amount of information analysed can be limited by considering a semi-local scale instead of a global one.

Local approaches are usually characterized by a lowest computational cost than global approaches. The local model-based approach from Shpilman and Brailovsky [5] uses a monodimensional accumulator array. A density map is first computed from the edge image and used to segment the image into high and low contour density areas. The line detection process is performed iteratively only in high contour density regions. At each iteration, an edge point is randomly selected and is locally analysed to determine the line direction. The line segment extremities are obtained using the segmented image and the detected line is deleted. The process is repeated until all the image has been analysed. The approach from Burns et al [2], known as line support region, relies on the definition of some specific regions, named line support regions, which contain a line segment. Here the input image does not consist of an edge map but rather of a gradient image. Edges detected can be combined in regions using gradient magnitude and direction of neighbouring pixels. The main line of each region is then obtained from a local statistical study of gradient magnitude in the concerned region.

In our comparison framework, we have chosen to consider not only classical detectors but also some original ones characterized by a very low computational cost, even when dealing with large color images.

4 Original approaches

As we are concerned with real time image analysis for scene understanding, we have also to consider low complexity techniques in our comparison framework. Here we present two original low-cost line detectors.

The first one called adaptative edge linking approach performs line detection using gradient magnitude and direction information. Here a pixel is considered to belong to a line if its gradient magnitude is higher than a threshold T but also if there exists at least a pixel in its neighbourhood which is characterized by similar gradient magnitude and direction values. Contrary to the original edge linking technique where T is a fixed parameter, here the value of the T depends on the length of the line which is currently being detected. The more points it contains, the higher the confidence of the process is, and the lower the threshold is. It allows the process to tolerate some noise in the line. However, this approach is purely local and is limited when lines contain holes due to the noise present in the images.

So we proposed another approach which relies on the connection of short line segments. Here we still consider gradient images instead of binary edge maps. Once again, we look for pixels characterized by high gradient magnitude. From each of these pixels P, we initiate a straight line creation process by selecting the neighbouring pixel with the highest gradient magnitude. It gives then a direction for a potential straight line, which is built until we obtain the last segment point. This process is iterated with another pixel with high gradient magnitude. Once all the image has been scanned, short line segments are available and have to be connected in order to avoid the approach to be purely local. More precisely, two line segments are connected if three conditions hold: the distance between their closest respective extremities has to be below a given threshold (which represents the process behaviour relatively to the image noise), the gradient magnitude of the intermediary points has to be non-zero, and finally the new line created should be a straight line. These three conditions allow the process to connect line segments which belong to the same straight line. In order to limit the sensitivity to noise, lines associated with a length too short or an average gradient magnitude too low are discarded.

We have presented here two new techniques for line detection. In the context of background modelling, we will now compare these approaches with some classical ones that have been presented in section 3.

5 Comparison of line detectors

In order to compare the presented approaches, we have to select a representative dataset and some appropriate evaluation criteria. We are to measure the efficiency and the quality of the methods (relatively to our goal, *i.e.* background modelling). As most of the line detectors rely on the use of a specific operator to compute gradient information, we have selected the Canny edge detector [3] as a common operator for all methods.

5.1 Testing dataset

The goal of the testing dataset is to be relatively heterogeneous to allow the comparison to take into account several criteria. These criteria are linked either with the content or the size of the images. Indeed, some image properties will have a direct influence on the quality of the results obtained. Depending on the line detector used, the image global contrast and the noise present in the images will affect the results. Moreover, some methods will deal more easily than others with images containing objects or background which have an irregular shape. So the images contained in the dataset have been selected to represent these different possibilities. The images are shown in the figure 2. We have also decided to include in our dataset graylevel and color images from different sizes. The properties of the images used are also given in the figure 2.

5.2 Efficiency evaluation

The efficiency is evaluated by analysing the processing time required to detect lines in an image. We have performed the line detection on the different images from the dataset using the approaches presented previously : Standard Hough Transform (SHT), Probabilistic Hough Transform (PHT), Combinatorial Hough Transform (CHT), Local Model-based Approach (LMA), Line Support Regions (LSR), Adaptive Edge Linking (AEL), and Connection approach (CON). The left part of table 1 illustrate the results obtained (in seconds) with a Dual Processor (Pentium III – 700 Mhz) PC. As we can notice in this table, the connection approach is the most efficient method with respect to processing time. The adaptive edge linking is also characterized by reasonably low processing times. Some methods (CHT, LMA, LSR) give interessant results but only when the image size is relatively small. As computation time includes memory allocation time which can be avoided when dealing with video sequences, line detection can be performed in real time using the connection approach or the adaptive edge linking.

5.3 Quality evaluation

As we are concerned with background modelling, the goal of the quality evaluation is to measure the ability of the different approaches to model correctly the background structure. This ability can be evaluated by comparing the result obtained with a reference result known theoretically. So, for every image in the dataset, we have performed a line detection manually in the context of background modelling.

Once the reference results have been obtained, it is necessary to compare them with results from line detection approaches. Let us note I_1 and I_2 the two images to be compared. An image I contains a number of lines noted $n(I)$ and the straight lines are indicated by an integer between 1 and $n(I)$. The goal is to match similar lines (i, j) from the two different images I_1 and I_2. We choose to measure the angle (noted $a(i, j)$) between the two lines and the shortest distance (noted $d(i, j)$) between two points of each line and we consider a weighthed sum of these normalized measurements with some coefficients α and β. These coefficients are set empirically depending on the context in which the comparison is performed. In our comparison framework, we assume angle and distance have similar importance and we set $\alpha = \beta = 1$. In order to match a line i from an image I_1 with some line in the image I_2, we search for the most similar line in I_2, which results in the definition of the following matching coefficient of a line i in I_1 with image I_2:

$$C_i(I_1, I_2) = \min_{j \in I_2} \left(\alpha \times \frac{a(i,j)}{\frac{\pi}{2}} + \beta \times \frac{d(i,j)}{\sqrt{h(I_1)^2 + w(I_1)^2}} \right) \tag{1}$$

where $h(I)$ and $w(I)$ represent respectively the height and the width of the image I. It is then possible to compute this coefficient for all lines from images I_1 and I_2. The global dissimilarity coefficient C is then :

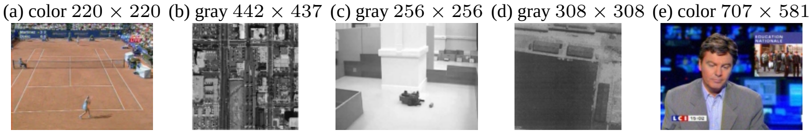
$$C(I_1, I_2) = \frac{\sum_{i \in I_1} C_i(I_1, I_2)}{n(I_1)} + \frac{\sum_{j \in I_2} C_j(I_2, I_1)}{n(I_2)} \tag{2}$$

For every image of the dataset, we have compared the results obtained using the different methods with the reference result defined manually. The global dissimilarity coefficients which have been measured are given in the right part of table 1. As we can observe, three methods are characterized by a higher quality than others : LMA, AEL, and CON. If we merge the efficency and quality evaluation results, the connection approach seems to be the best approach for line detection. Other methods of interest are the adaptive edge linking and the local model-based approach.

6 Conclusion

In this paper, we were dealing with background modelling to achieve scene understanding. Among a set of different kinds of features, we have shown that straight lines can be appropriate to model the scene background efficiently. We have then compared several classical line detectors from the literature with two new very low-cost approaches, using a heterogeneous dataset and two measures to evaluate the efficency and the quality

(a) color 220×220 (b) gray 442×437 (c) gray 256×256 (d) gray 308×308 (e) color 707×581



(f) color 320×240 (g) gray 410×407 (h) gray 320×240 (i) gray 304×304 (j) color 320×240

Fig. 2. Images used in the comparison of line detectors.

Number	SHT	PHT	CHT	LMA	LSR	AEL	CON
(a)	7.34	3.70	**0.06**	**0.06**	1.66	0.55	0.11
(b)	41.50	23.39	0.97	0.54	5.59	2.89	**0.41**
(c)	32.78	26.83	0.91	1.16	0.64	0.41	**0.16**
(d)	57.72	44.45	2.19	1.84	1.17	0.22	**0.19**
(e)	131.05	101.17	26.13	16.16	15.55	3.88	**2.02**
(f)	39.74	37.88	1.53	1.61	1.34	1.00	**0.61**
(g)	67.70	37.98	4.75	6.05	1.84	2.23	**1.00**
(h)	35.36	20.77	0.73	0.89	1.70	0.28	**0.25**
(i)	44.00	25.00	1.72	2.30	0.72	**0.17**	0.45
(j)	43.58	20.25	1.27	2.19	1.36	**0.59**	**0.59**
average	129.74	94.43	4.09	4.32	5.42	2.44	**1.22**

Number	SHT	PHT	CHT	LMA	LSR	AEL	CON
(a)	46.66	58.50	40.39	26.07	24.22	18.71	**16.20**
(b)	31.54	46.63	41.78	**14.16**	45.35	23.92	16.12
(c)	63.21	65.64	73.39	47.39	69.32	70.64	**26.77**
(d)	39.84	43.78	44.22	23.47	24.35	**20.03**	**20.03**
(e)	53.20	41.32	83.37	**32.25**	53.66	32.97	42.65
(f)	30.37	28.23	38.21	20.72	37.94	**18.13**	25.48
(g)	43.94	42.93	52.74	55.68	44.66	**23.20**	42.87
(h)	25.71	24.98	40.63	31.04	28.53	**17.11**	17.87
(i)	53.52	52.83	53.25	**33.75**	65.80	98.47	49.11
(j)	76.10	69.88	85.49	58.26	95.24	**51.21**	51.36
average	44.17	43.76	53.74	33.63	43.02	**32.33**	33.22

Table 1. Comparison results: (left) relative computation times (in seconds) and (right) quality measured for line detectors.

of the line detectors. From the preliminary results we obtained, a contributed method called connection approach gives the best results.

In order our comparison to be more exhaustive, we have to consider other line detectors and other quality and efficiency measures. Efficiency evaluation can take into account the possible implementation of the algorithm on a multiprocessor workstation whereas quality evaluation can be based on concepts from human vision theory.

References

1. D. Ben-Tzvi and M.B. Sandler. A combinatorial Hough Transform. *Pattern Recognition Letters*, 11(3):167–174, March 1990.
2. J.B. Burns, A.R. Hanson, and E.M. Riseman. Extracting straight lines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(4):425–455, July 1986.
3. J.F. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.
4. G. Roth and M.D. Levine. Random sampling for primitives extraction. In *International Workshop on Robust Computer Vision*, pages 352–366, Seattle, USA, October 1990.
5. R. Shpilman and V. Brailovsky. Fast and robust techniques for detecting straight line segments using local models. *Pattern Recognition Letters*, 20(9):865–877, September 1999.