POLISH ASSOCIATION FOR SPATIAL INFORMATION
ANNALS OF GEOMATICS 2008 ❍ VOLUME VI ❍ NUMBER 2

EXTRACTION OF VEHICLE CLASSES FROM A DIGITAL CAMERA OBSERVATION FIELD

EKSTRAKCJA KLAS POJAZDÓW Z OBRAZU POLA OBSERWACJI KAMERY CYFROWEJ

Jan Piecha, Marcin Staniek

Department of Informatics Systems in Transport, Faculty of Transport,
Silesian University of Technology, Poland

Keywords: video detection, image processing, on-line traffic control, vehicles classes
Słowa kluczowe: wideo-detekcja, przetwarzanie obrazów, sterowanie ruchem on-line, klasy pojazdów

Introduction

Instead of counting passing vehicles, a video image of the vehicles can be analysed. Video cameras provide a very rich data source that can be used for many purposes, including counting vehicles and identifying their type, calculating vehicle speed, and accident detection. The image analysis can be carried-out by both programming techniques (Kpalma and Ronsin 2001, pp. 170) and hardware solutions (Iannizzotto et al., 2001, pp. 100). The programming technologies offer a wider range of solutions than hardware image processing units, however, programming solutions provide poor performance compared with an real-time controlling unit (Smith, 1998). Correctly defined image processing algorithms enable the reduction of redundant data, facilitating effective traffic monitoring and control. In this work, segmentation algorithms (Tadeusiewicz and Korohoda, 1997, pp. 243) were improved through a combination of image space division, space growth of the image, edge detection of the image segments, and static segmentation of the image. This refinement of the source image retains adequate information to achieve interactive real-time analysis and effective performance.

Image segmentation process

The segmentation process allows the computer to extract objects from the image according to specific criteria. Alongside the extracted features, one can distinguish other features including colour composition, pixel brightness levels, and object shape. The segmentation process distinguishes objects and defines the image background. The analysed pixel values are compared with the given threshold value of each attribute. The most difficult problem concerns this threshold value. In order to make the real-time vehicle classification process faster, image segmentation was used to preprocess the data, which leaves the extracted segments described by their characteristic features. The image converter produces decimal attributes assigning the images, in accordance with the segment's current number. In this method the pixel's brightness attribute is compared with neighbouring pixels. Pixels are combined when their attributes are the same or almost the same. It is an expansion process of the object's surface, with the same attributes. The fundamental advantage of the expansion segmentation algorithm is its noise filtering (threshold comparison) ability. The segmentation method (i.e. edge detection) extracts situations where different pixels exist in the image, following which the image is recorded in the form of a binary map.

The extracted edges have to be written as a closed line, so each separate object image segment can be assigned. Bad quality images need more advanced edge detection algorithms, typically using differential equations. Statistical segmentation algorithms work on homogeneous image fragments and produce satisfactory results for regular objects; but are not recommended for detecting vehicles.

Pixels indexing

Figure 1. A source image of the road scene

In image analysis the homogeneous fragments are assigned (i.e. labelled) by indexes of the image segment current number.

The procedure runs differently for differently shaped geometrical objects (convex or concave) (Tadeusiewicz and Korohoda, 1997, pp. 252). The segmentation method introduced in this paper uses products of the image filtering algorithm that produces maps of object edges. Figure 1 shows the source image of the road scene. This image is processed in several steps: noise filtering; object edge detection; and threshold value filtering.

Noise reduction removes extremes of pixel brightness by reference to the linear median. This operation is free of image quality destruction, as by convolution filters.

The edge recognition algorithm calcu- lates the gradient of the image intensity using the Sobel operator, which uses two orthogonal masks as in Figure 2.

In successive steps, comparison of the gradients of two directions produces the layout of the edges in the image. The algorithm simplification by applying a modular formula (1):

$$M_{i,j} = \left|M_{H\,i,j}\right| + \left|M_{V\,i,j}\right| \tag{1}$$

where: $M_{Hi,j}$ – lightness of the pixel attribute value for horizontal edges,
$M_{Vi,j}$ – lightness of the pixel attribute value for vertical edges,
$i = 0...n - 1,\ j = 0...m - 1.$

Image binarisation, is executed after the image edges are defined. This procedure results in a significant reduction of the image file size. The threshold for the elaborated method is achieved by using two values from the image gray scale, from the function (2) of the lightness medium square:

-1	0	1
-2	0	2
-1	0	1

-1	-2	-1
0	0	0
1	2	1

Figure 2. Orthogonal masks of the Sobel operator

$$p = \sqrt{\frac{\sum_{i=0..n-1} \sum_{j=0..m-1} {M_{i,j}}^2}{n \cdot m}} \quad (2)$$

where: $M_{i,j}$ – lightness of pixels with co-ordinates indexes $\langle i, j \rangle$
n – is width and m – is height of the camera observation field.

The threshold value is used for data binarisation of the image and its edges (Fig. 3) that describe the source data unit (3):

$$P_{i,j} = \begin{cases} 0 & for \quad M_{i,j} \geq p \\ 1 & for \quad M_{i,j} < p \end{cases} \quad (3)$$

where: $i = 0...n-1$, $j = 0...m-1$,
n – width, m – height; of the camera observation field.

For the elaborated method, a segment fulfillment provides additional results with edges of the object's assignment.

The fast algorithm for object assignment is used for filling the image areas bounded by closed curves. The internal pixels in these extracted objects are denoted by the same grey-scale attributes.

The implementation of Smith's algorithm

Smith's algorithm, the most effective method of various algorithms for image segmentation, with simple procedures and fast extraction of objects, was applied. Using the source image shown in Figure 1, a map of edges was produced, where the image is re-coded to two values of pixels $P_{i,j}$:

$$P_{i,j} = \begin{cases} 1 & for \quad pixels \quad on \quad edges \quad (black \quad colour), \\ 0 & for \quad object \quad - \quad piksels \quad (white \quad colour). \end{cases} \quad (4)$$

where: $i = 0 \ldots n-1$, $j = 0 \ldots m-1$, n – width, m – height of the camera observation field.

The image segmentation procedure gives the image areas description, shown in Figure 4. Each ($P_{i,j} = 0$) pixel is placed within the internal boundaries of the object's map.

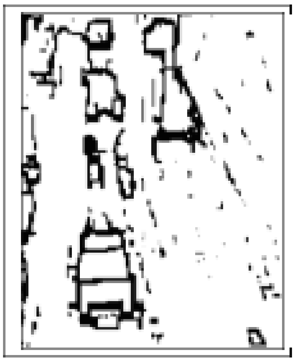

Figure 3. The image edges assignments

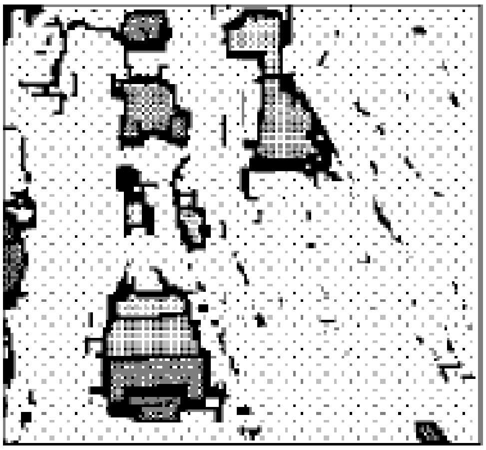

Figure 4. The image after segmentation

The internal surface of the object is assigned by its current decimal number *(k)*. The pixel value equal to one is assigned to the object's edge. In Figure 4 the image resulting from using the segmentation process is shown. There, textures assign the extracted objects/segments. The segmentation procedure starts from the image checkup, at a specific starting point of the object, as illustrated below. The first pixel is denoted by a value of zero and coordinates $\langle i, j\rangle$. It describes the starting point (Fig.5), from where the scanning procedure and filling-up of the object begins (5):

$$P_{i,j} = 0 \quad \rightarrow \quad x_C = \mathrm{i}, \quad y_C = j \tag{5}$$

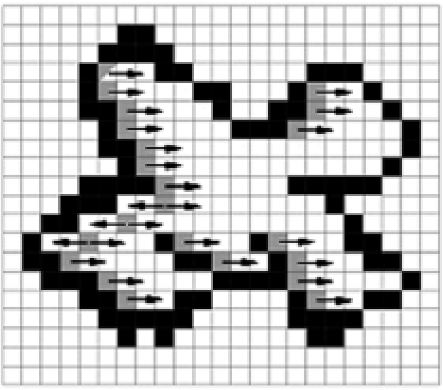

Figure 5. Starting points (grey squares) $P_{x_c,\, y_c}$ assignment

After this step, the program returns to the starting point $\langle i, j\rangle$ and the next area of the object is processed. After this is completed, the program searches for the edge of the next object. The starting pixel belongs to the specific pixel set of the map of image edges.

The starting point for the scanning procedures uses two triangles (i.e. white and grey). In the scanning algorithm, the two procedures are different: the right sided area starts at the pixel from the L_{SP} line, from the left side, limited by the object edge, where $P_{x_c-1,\, y_c} = 1$. The two sided scanning procedure was assigned by the line L_{SO}, for which $P_{x_c-1,\, y_c} = 0$.

The right sided scanning L_{SP} of the segment k is defined by equation (6):

$$L_{SP} = \left\{ x, y : \underset{x_C \le x < x_K}{\forall} P_{x,y_C} = 0 \cap P_{x_K,y_C} = 1, \quad y = y_C \right\} \tag{6}$$

where: x_c, y_c – coordinates of the starting pixel, x_k – coordinate x that limits the right sided part of the object field of scanning.

The right sided scanning procedure is illustrated in Figure 6. The grey square (pixel) k, assigns a current number (decimal value – the filler) for the scanned object.

The two sided scanning consists of phases: the right-sided scan begins from the starting pixel-point P_{x_c, y_c} and the left-sided scan starts from the pixel $P_{x_c, -1y_c}$. These processes use the pixels set on two sides of the scanning starting points (sets of pixels L_{SP} and L_{SL}): $L_{SO} = L_{SP} \cup L_{SL}$

The left-sided scanning raw is described by (7):

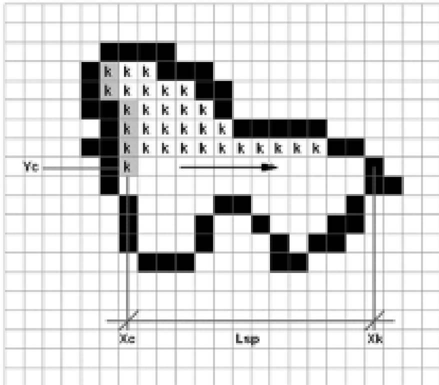

Figure 6. The right sided L_{SP} scanning

$$L_{SL} = \left\{ x, y : \forall_{x_P < x \le x_C - 1} P_{x_P, y_C} = 1 \cap P_{x, y_C} = 0, \quad y = y_C \right\} \tag{7}$$

where: x_c , y_c – are the starting pixel coordinates, x_p – x coordinate's limit of left sided scanning.

Objects scanning algorithm

The procedure for the scanning of the row attributes in every row L (i.e. for right-sided scanning L_{SP} or two-sided L_{SO}), consists of two phases. The first phase concerns assignment of the object's internal pixels value k:

$$\forall_{x,y \in L} P_{x,y} = k \tag{8}$$

where: k is a current number of the image segment.

The second phase (9) concerns the identification of the starting pixels set identification, for pixels coordinates $\langle x, y \rangle$ in each row L.

$$P_{x,y+1} = 0 \cap P_{x,y} = k \cap \left(P_{x-1,j} = 1 \cup P_{x-1,y+1} = 1\right), \qquad \langle x, y \rangle \in L \tag{9}$$

The next step concerns scanning the row below, commencing from the starting pixel at coordinates $\langle x, y + 1 \rangle$. Pixels in this row also belong to the same segment k.

The above case assigns the starting pixel at coordinates $\langle x, y - 1 \rangle$, above the previous row in segment k.

$$P_{x,y-1} = 0 \cap P_{x,y} = k \cap \left(P_{x-1,j} = 1 \cup P_{x-1,y-1} = 1\right), \qquad \langle x, y \rangle \in L \tag{10}$$

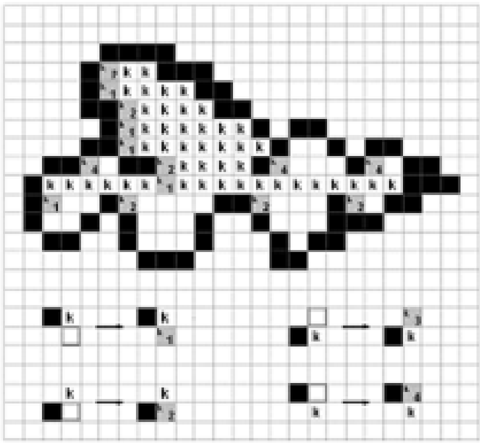

Figure 7. The scanning procedure example

The scanning procedure principles are illustrated in Figure 7. Each gray pixel gives the starting point of the row. The scanned object filler *k* corresponds to the object number, used as an identifier.

The gray square *p* is a starting point of the filling-up procedure. White pixels are not identified – they do not belong to the segment. The black pixels indicate the object edges.

The segmentation algorithm is described by the pseudo code:

For i = 0..n-1 and j = 0..m-1
IF $P_{i,j}$ = 0 (the pixel of the scanned field) THEN
k ← k + 1 {increment segments counter}
Send into the ***table*** $\langle i, j\rangle$
UNTIL ***table is not empty*** execute: ***fetch from the table*** $\langle x_C, y_C\rangle$
$x \leftarrow x_c,\ y \rightarrow y_c$ UNTIL $P_{x,y}$ (the image pixel) execute
The field scanning $(x, y, 1)$ IF $P_{Xc-1,\,Yc}$ = 0 THEN
$x \leftarrow x_c - 1$ UNTIL $P_{x,y}$ = 0 (the field pixel) DO
Scanning of the field $(x, y, -1)$

Procedure ***Field Scanning*** *for coordinates (x, y, q)* is executed traditionally, for every pixel ($P_{x,y}$), in three steps:

1. pixels assignment in accordance with the segment number (8).
2. starting pixels assignment in neighbor image segments, according to equations (9) and (10).
3. increment or decrement of pixels' coordinates *x* with relation to the scanning direction as it is assigned by *q*; scanning directions.

Image Parameterisation

Parameterisation of the image defines specific characteristic features of the image. It uses the assigned dimensions of the object classification of vehicle class and the object's characteristic features comparison (the extracted object's weight values finding). The parameterisation algorithm distinguishes between local (i.e. geometrical descriptors) and global (i.e. number of segments in a surface unit) characteristic values of segments (Płaczek and Staniek, 2007). If characteristic features are properly defined then effective classification of image objects is made possible.

Local descriptors

Surface S, is defined by the sum of pixels in segment: $P_{i,j}$ in segment k:

$$S(k) = \sum_k P_{i,j} \tag{11}$$

Length of the segment edge, is defined by number of pixels on the object's edge:

$$L = \frac{\pi}{4} \cdot \left[\cdot a \cdot \left(N_0 + N_{90} + \frac{a}{\sqrt{2}} \cdot (N_0 + N_{90}) \right) \right] \tag{12}$$

where: N_0, N_{45}, N_{90}, N_{135}: the segment projection, a – a distance between image points.

This offers a new definition of the length on the segment edge periphery L, as a set of pixels pairs on the segment k edge. The set defines the length of the edge as:

$$L(k) = \sum_k \{ P_{i-1,j} = 1, \quad P_{i+1,j} = 1, \quad P_{i,j-1} = 1, \quad P_{i,j+1} = 1, \} \tag{13}$$

The Fereta diameter (D_v, D_h) defines the linear parameters of the segment size – vertical v and horizontal h of segment k in pixels $P_{i,j}$ (Fig. 8):

$$D_v = \max(i) - \min(i) \text{ for } P_{i,j} = k \qquad D_h = \max(j) - \min(j) \text{ for } P_{i,j} = k \tag{14}$$

The centre of gravity describes the object location. It is found by an object's moment of torpidity M_i, M_j, for a surface value $S(k)$:

$$M_i = \frac{1}{S(k)} \sum_k i_k \quad M_j = \frac{1}{S(k)} \sum_k j_k \tag{15}$$

The projection lengths, for image parameterisation, are described by measures of segment (I_z, I_p) for full size (the object's length and Feret diameter) in any defined length. The projection length was defined as the biggest distance between projections, as indicated in Figure 9.

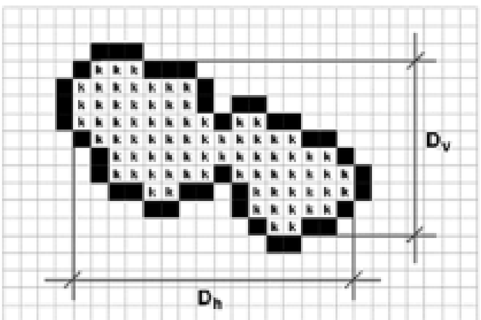

Figure 8. Example of segment k Feret diameter

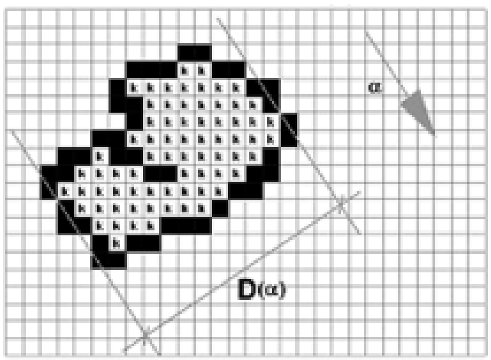

Figure 9. The object projection length

The object shape coefficient

The shape coefficients are defined (Piecha and Staniek, 2007) by one of several methods. Fast methods are recommended for the recognition of real time objects parameters. A second group defines more complex problems; although more precise for the object description, they are slower. For a real-time implementation the first group of algorithms was chosen (Piecha and Staniek, 2007), which use the object R_z for content coefficient recognition and the Feret coefficient R_F measure.

The object shape and content R_z of the image segment was defined as:

$$R_z = \frac{L^2}{4\pi S} \tag{16}$$

where: S – surface of the segment, L – the segment periphery.

Elongation of R_F the object:

$$R_F = \frac{D_h}{D_v} \tag{17}$$

where: D_h, D_v – the Feret vertical and horizontal diameters, respectively.

For these image parameterisations, the list O_k of these objects was constructed:

$$O_k = (x_s, y_s, L, S, I_z, I_p, R_z, R_F), \tag{18}$$

where: the object k current number, $\langle x_s, y_s \rangle$ – the object gravity centre, L – the segment periphery, S – surface size, I_z, I_p – projection length, R_z – content coefficient, R_F – the Feret coefficient.

Final remarks

Figure 10. Final image example after selection procedures

The Smith's algorithm significantly simplifies the parameterisation process. It uses surface, periphery, Feret diameters and gravity centres of the segment, which produces the parameterisation closest to the shape parameters of the objects, i.e. the coefficient of dimension R_z and elongation R_F.

At the beginning of the selection process small dimension objects are erased. In the next step bigger objects from the threshold surface are deleted. Finally the extracted image (Fig.10) is then further analysed. The classification procedures find the objects that fulfill threshold levels of all the specified characteristic measures; for example threshold S_p = 100 pixels and shape coefficient $R_{z\text{-}pr} = 15$. With a white background, this method allows us to avoid coding the background and, consequently, considerably reduces the memory size needed for the image description. This results in a reduction in complexity and computation time for the image analysis.

References

Iannizzotto G., Lanzafame P., La Rosa F., 2001: A Real-Time Solution to the Image Segmentation Problem: CNN-Movels, In Obinata G., Dutta A. (eds) Vision-Systems Segmentation and Pattern Recognition, I-Tech Education and Publishing, Vienna, Austria, 2007, pp. 099-116.

Kpalma K., Ronsin J., 2001: An Overview of Advances of Pattern Recognition Systems in Computer Vision, In Obinata G., Dutta A. (eds) Vision-Systems Segmentation and Pattern Recognition, I-Tech Education and Publishing, Vienna, Austria, 2007, pp. 169-194.

Smith S.M., 1998: ASSET-2: Real-Time Motion Segmentation and Object Tracking, *Real-Time Imaging*, Vol. 4.1, pp. 21-40.

Tadeusiewicz R., Korohoda P., 1997: Computer Analysis and Image Processing (in Polish), Foundation of Progress in Telecommunication, Cracow, ISBN 83-86476-15-10.

Płaczek B., Staniek M., 2007: The vechicles tracking algorithm for hardware video-detection implementation, Transactions on Transport Systems Telematics, Modeling, Management and Image Processing, Monohraph edited by Jan Piecha, Silesian University of Technology, Gliwice, pp. 120-127.

Piecha J., Staniek M., 2007: The vehicle description method by means of its extracted image. Proc. of Int. Conf. "Transportation and Logistics Integrated System", AGH University of Science and Technology, Institute of Automatics, Cracow, pp. 204-215.

Abstract

For graphic object analysis of images recorded by a digital camera, several image recognition algorithms were investigated. In addition, investigations for an effective interactive real-time digital video-image vehicle stream classification has been undertaken. The researched classification methods concern criteria requiring a calculation of small complexity, which produces a high quality classification. The proposed image segmentation approach provides a technology that allows the user to find in the specified image its features characteristic for vehicle classes using efficient processes.

Streszczenie

Zbadano kilka algorytmów ekstrakcji obiektów z obrazów cyfrowych, które mogą być zastosowane do klasyfikacji pojazdów rejestrowanych kamerą cyfrową. Badania dotyczące metod klasyfikacji pojazdów koncentrują się na algorytmach zapewniających niewielka złożoność obliczeniową oraz wystarczająca dokładność. Zaproponowana metoda segmentacji pozwala znaleźć w obrazie cechy kształtu obiektów charakterystycznych dla danych klas pojazdów.

prof. dr hab. inż. Jan Piecha
jan.piecha@polsl.pl

mgr inż. Marcin Staniek
marcin.staniek@polsl.pl