

A SIMPLIFIED AND FAST VERSION OF THE HUECKEL OPERATOR FOR FINDING OPTIMAL EDGES IN PICTURES

L.Mérő - Z. Vassy
Computer and Automation Institute,
Hungarian Academy of Sciences
Budapest, Hungary

Abstract

A simplified version of the Hueckel operator was worked out to preserve all the advantages of the original one with a running time considerably shorter. Its concept is based on the reinterpretation of the Hueckel operator as a template matching method. This reinterpretation renders some theoretical and computational simplifications possible. As a result, an optimal edge-shaped template is constructed by linear combination of two suitably chosen original ones. The parameters of this optimal template characterize the edge segment found in a picture area.

Introduction

Hueckel's operator for edge detection in pictures is one of the best tools which have been invented for this kind of tasks. In very noisy pictures its high effectiveness can be comparable only with the global optimization procedures e. g. by Montanari and Martelli (8), (9), but without their drawbacks in processing time and memory requirement. The Hueckel operator processes a picture block containing 52 points on PDP-10 in 10 msec, e. g. if it scanned for example a 256 by 256 point picture so that the blocks overlap by one thirds of each other, the processing time would be about 40 seconds. Of course, by means of sophisticated edge-following methods this time can be reduced significantly, thus the applicability of this operator is not beyond the possibilities to process a real picture. These 10 msec are, however, rather large to look for some modifications based on its basic idea to speed up the operation, even by possible simplifications which preserve the excellent properties of the original one.

In this paper we present a method aiming at this by reinterpreting the principle of the Hueckel operator and giving a modification of it based on our interpretation.

Hueckel's method in its original form

The original interpretation of the Hueckel operator can be summarized as follows:

- a/ We seek for a function S on a D area in the input picture, which function has a constant b value on one side and

another constant $b+d$ at the other side of a certain line, the equation of which is

$$cx + sy = r;$$

- b/ This function S is determined by the least squares method, i. e. we minimize the sum of the square of differences point by point inside of D between S and the input $F(x,y)$ gray level function;
c/ For the sake of easy computation we set S as an infinite Fourier series of some H_1 base functions;
d/ The infinite set of these H_1 functions is approximated by a finite set of its lowest frequency members $H_1 \dots H_7$;
e/ We complete the minimization task in this new space. The parameters of S characterize the parameters of the line segment found in D .

Hueckel extended this method for finding not only edges but lines as well (6), but this extended version will not be treated here, as we are interested in edge detection only. The sketches for $H_0 \dots H_7$ are seen in Figure 1; + and - marks denote the sign of the functions, the lines inside their area D denote the zero crossings. We also do not present other details, though some of them /e.g. the so-called "solution theorem"/ are even theoretically very attractive, too.

In the following we will prove that the Hueckel operator can be viewed as a kind of template matching method. Before this we are to briefly survey some approaches to template matching.

Template matching

Let be given two function, $f(x,y)$ and $g(x,y)$ on a G two-dimensional area, and let us seek $f(x,y)$ in $g(x,y)$, i. e. the area in G where g is identical to f . In this case we can compute the following convolution:

$$\int_{H \in G} f(x,y) g(x,y) dx dy = f \otimes g$$

and the sought area H' is just where this convolution takes the maximum value in G . $f(x,y)$ is called template function or simply template (13).

If $f(x,y)$ is edge-shaped, it can be used to find edges in pictures. Rosenfeld (14), Rosenfeld and Thurston (15), (16)

and others (7), (10), (11), (12) have worked out several versions of this method and investigated them empirically. For example, perhaps the simplest templates for two-dimensional pictures are those of Shirai (17) sketched in Fig. 2. The first of them shows a maximum if it lies exactly on a vertical and the second on a horizontal edge. For making the method less noise-sensitive, one can take into account larger areas, e. g. 5x5 or 8x8 etc.

Since Hubel and Wiesel (4) pointed out that the visual information processing system in mammals operates in a way very close to this method, at least for the local edge and spot extraction on the retina, template matching has a certain theoretical support. Moreover, just this physiological achievement led Deutsch to develop his template called SLEN /Short Line Extractor Neuron (1)/ used for numerical character recognition. For real-time processing of real pictures, however, these methods do not produce so effective results as other ones, being though ad hoc without any theoretical suggestions. In the case of edge detection, if the shape and orientation of the template are fixed, the probability of good matching is little because the output value of the convolution decreases very sharply either because of the increase of the angle between the template edge and the real one or the curvature difference of them (2). From this two main shortcomings follow. First, the system would not find certain, un-luckily oriented edges even from the line-shaped ones. To surmount this difficulty, Holdermann and Kazmierczak (3) have presented a method with parallel application of several templates to the same picture area around a point. They took care also of noise elimination by averaging, and their output picture after this multiple template operation looked quite good in quality, but even their method could not avoid the following - second - difficulty: in the case of fixed templates the large majority of the required computer operations falls into the category of unsuccessful trials and the whole processing time may be too long, especially if the picture is not "angular" enough. The main reason of these shortcomings is obviously the rigidity of the templates applied. In contrast to e. g. Holdermann and Kazmierczak's eight templates, we would need only one of them; but that would be the best matching one. The question is how to find just this "optimal template", i.e. how to solve the opposite problem to that of template matching: fitting the template to a fixed picture area and not the picture area to a/some fixed template/s/. Let D be this area of the picture, τ the set of all possible templates, and $F(x, y)$ the input two-dimensional grey-level function, then we seek $T(x, y/\epsilon\tau)$ so that

$$\int_D T(x, y) F(x, y) dx dy = T \otimes F = \max. \quad (1)$$

In the next section we will point out that Hueckel's operator completes just this task.

The Hueckel operator as a kind of template matching method

We will first need to recall some of Hueckel's notations. Let the grey values of the input picture on a circle-shaped area D be $F(x, y)$, the sought two-dimensional step function be

$$S(x, y, c, s, r, b, d) = \begin{cases} b & \text{if } cx + sy > r \\ b+d & \text{if } cx + sy < r \\ 0 & \text{otherwise.} \end{cases}$$

Let the base functions chosen in the Hilbert space of all the functions on D be $H_0(x, y), \dots, H_7(x, y)$.

With these notations the functional equation to be solved is:

$$\int_D (F(x, y) - S(x, y, c, s, r, d, b))^2 dx dy = \min. \quad (2)$$

The Fourier coefficients for F and S are:

$$f_i = \int_D F(x, y) H_i(x, y) dx dy \quad (3)$$

$$s_i(c, s, r, d, b) =$$

$$= \int_D S(x, y, c, s, r, d, b) H_i(x, y) dx dy \quad (4)$$

$$\lambda(c, s, r, b, d) =$$

$$= \sum_{i=0}^7 (f_i - s_i(c, s, r, b, d))^2 = \min. \quad (5)$$

Hueckel's operator solves this functional equation^x. Now we will prove that /5/ can be rewritten in the form of /1/ in the previous section. For this we need a lemma:

Lemma: Let

$$\phi_i \quad /i=1, \dots, N/ \text{ and } \psi_i/p/ \quad /i=1, \dots, N/ \text{ so}$$

that

$$\sum_{i=1}^N \psi_i/p/ = \text{const.}$$

where p is a vector of arbitrary dimensions.

If

$$u/p/ = \sum_{i=1}^N (\phi_i - \psi_i/p/)^2 \quad \text{and}$$

$$v/p/ = \sum_{i=1}^N \phi_i \psi_i/p/$$

^xHere we neglect Hueckel's heuristic weighting coefficients as they have no relevance to the problem considered.

then if for some values of p

$\mu/p = \min.$ then for the same
values of p
 $\nu/p = \max.$

The proof is given in Appendix A.
According to this lemma, /5/ can be re-written as

$$\lambda'/c,s,r,b,d/ = \sum_{i=0}^7 s_i/c,s,r,b,d/ f_i = \max.$$

Substituting f_i from /3/:

$$\begin{aligned} \lambda'/c,s,r,b,d/ &= \\ &= \sum_{i=0}^7 s_i/c,s,r,b,d/ \cdot \int_D F/x,y/H_i/x,y/dxdy = \\ &= \max. \end{aligned}$$

With some trivial transcriptions:

$$\begin{aligned} \lambda'/c,s,r,b,d/ &= \\ &= \int_D F/x,y/ \left[\sum_{i=0}^7 s_i/c,s,r,b,d/H_i/x,y/ \right] dxdy = \\ &= \max. \end{aligned}$$

The expression

$$\begin{aligned} T/x,y,c,s,r,b,d/ &= \\ &= \sum_{i=0}^7 s_i/c,s,r,b,d/ H_i/x,y/ \end{aligned}$$

is a template by its role in /6/, as the form of /6/ is identical with that of /1/. $T/x,y,c,s,r,b,d/$ is the optimal template to the $F/x,y/$ input function constructed as a linear combination from the set of original templates $H_0 \dots H_7$.

The possibility of simplifying the Hueckel operator

In Hueckel's interpretation the basic aim, i. e. finding an edge, is reached by fitting a two-dimensional function to the input grey-level function in a picture area. Consequently, a relatively large set of both the base functions and edge parameters is required, because it is obvious that no satisfying degree of fitting could be approached by a remarkably simpler base field and parameter vector /containing e.g. two or three functions or values respectively/. If we are, however, considering the method as a template matching one, the fitting problem becomes unimportant and its requirements cannot be taken into account.

First, we may choose a two-dimensional parameter vector $/c,s/$ instead of $/c,s,r,b,d/$ as the grey values of the two sides of the step function are irrelevant, and the exact location of the edge inside the D area can be computed in a very simple way once we have determined the edge direction. The direction is characterized only by c and s , as in the equation of the

edge

$$cx + sy = r$$

c and s are just the direction cosines to the $+x$ axis.

Second, if we need only the edge direction, we may work with only two of Hueckel's original templates $H_0 \dots H_7$, because it seems well even intuitively /we will prove it later, of course/ that by the linear combination of H_2 and H_3 the direction of the optimal template can be determined. By the direction of a template we mean the direction of the O -crossing of a H_2 or H_3 type function.

Third, if we do not take care of the mathematical conditions of the Fourier function fitting in a Hilbert space - as these conditions determined the choosing of $H_0 \dots H_7$ in Hueckel's work /Chapter 2/ - the only criterion in choosing our two templates can be the speed of computation which is very crucial in real applications. Thus we can use both constant-valued template functions and a more convenient shape for the D picture area. The convenience of this shape means also the speed of computing c and s or any other quantities characterizing the direction.

The method we have got to by the above considerations will be set forth in the following section. It is essentially simpler and faster than that of Hueckel, but as will be seen from the examples presented later, it preserves all of its advantages in comparison to other methods. The computation consists of two steps. The first one is the determination of the edge direction by constructing an optimal template from two beginner ones by linear combination. Just like in Hueckel's method, a vast part of the extremization problem is solved analytically and the number of the required arithmetical operations is small. The second step is the determination of the exact location of the edge in D by its two border points. It is solved by some simple logical rules and point reflections.

Description of our edge detection operator

The core of the method is based upon the following elementary theorem:

Theorem: Let D denote the unit square, f_1 be the function being $+1$ above the sub-diagonal of D , and -1 below it, f_2 be the function being $+1$ above the principal diagonal of D and -1 below it, /see Fig. 3, a, b/ and let f be a function being $b+d$ above a line e in D , and b below this line. /See Fig. 3, c or d/. Then if α denotes the angle between the subdiagonal of D and e , the following equation will hold:

$$\frac{\int_I f f_2 dx dy}{\int_I f f_1 dx dy} = \operatorname{tg} \alpha$$

The proof of the theorem will be presented in Appendix B. The odd thing about the theorem is that the expression at the left side does not depend either on b and d or on the real place of the line but its angle.

By this theorem we are given a very pleasant tool to determine an edge in a little part of the picture. The two integrals can be very easy and fast computed, and because of the integrations this method is of very low sensitivity for noises. /Later we shall elaborate further considerations about this./

In our work we have chosen D to be a square consisting of 8×8 picture points. Having supposed that there was a straight edge in the picture at D , by using the above mentioned theorem we could determine its place, i. e. its two endpoints in D .

To begin with we transform the value of the tangent to a more treatable form. We determine from the value of $\operatorname{tg} \alpha$ how many picture points are the difference between the right and the left end of a line in D , the angle of which with the subdiagonal is α . /It is enough to deal with lines lying in the quarter of the plane shown on Fig. 4, because if the edge lies in the other quarter, then by counting with $|\operatorname{tg} \alpha|$, after the whole process a reflection of the edge to the principal diagonal of D - i. e. its endpoints - gives the real edge./

Everything stated until now is valid both for pictures with two and for those of more grey levels. Now we shall show the determining process of the real edge at first only for pictures with 2 grey levels and after that we shall show how to generalize that idea for pictures with more grey levels.

In the case with 2 grey levels the sum of the 1's in D divided by 8 shows, roughly spoken, up to what a height D is filled with ones. This means e. g. that if the edge were horizontal, then the edge could be either the horizontal line at that height, or that reflected to the centre of D /for D could be filled with ones from its bottom or from its top/. But even if the edge were not horizontal, we could know what was the difference between its left and right endpoints' height, so we could determine the edge. If both the left and the right endpoints of this edge were inside of D then we had only to determine whether this was the real edge or we had still to reflect it to the centre of D .

If one of the endpoints of the above obtained edge were not in D , it could be only by 1, 2, or 3 points sticking out of D , because now already the edge we are working with lies in the plane quarter shown on Fig. 4. But in these cases it is already very easy to determine the real edge.

For pictures with more grey levels we could use the same method if we knew b and d , because then the number of the points with greater grey level could be approximated by the following expression:

$$s' = \frac{S - 64b}{d}$$

where S denotes $\int_I f dx dy$, i. e. the sum of the grey levels of all points of D . But now by using the notations of Fig. 5, let T_1, T_2, T_3 and T_4 denote the sum of the grey levels of f /i.e. of the picture/ in these 4 quarters of D , and $\frac{\min\{T_i\}}{16}$ resp. $\frac{\max\{T_i\}}{16}$ approximates well b

resp. $b+d$. After this the above described method of finding the edge can be followed in the case with more grey levels as well.

By this method we have already found either the real edge in f or the same reflected to the centre of D . /See Fig.6/ To determine which of them is the real edge, the following table shows a method provided that up to this time we have supposed in the steps that D was filled by the ones /resp. by $b+d$'s/ from its top. The case A means that we have found the real edge while B means that still we have to reflect our edge to the centre of D .

If	$\operatorname{tg} \alpha \geq 0$	$T_1 \geq T_2$	then A
		$T_1 < T_2$	then B
		$T_3 \geq T_4$	then B
	$\operatorname{tg} \alpha < 0$	$T_3 < T_4$	then A

This method was implemented by a program on the assembly language of a 16 bit microprogrammed minicomputer of 550 program instructions without any cycles. If there is an edge in D it needs 250-400 machine operations, i. e. about 500-800 μ secs. No floating point arithmetics are needed.

About the reliability of the method rough under-estimations show that supposing a noise with a binomial distribution with a mean value of 5 error points in a 8×8 square, the probability of the difference between the edge determined by this method and the real edge being not greater than 1 point anywhere, is over 95 %.

Finally we present three examples for the working of the method. Example 1 is the example shown in Hueckel's article (5)

which we have completed at the corners. The dotted line shows the edge found by Hueckel, the continuous one is the edge found by our method.

Appendix A

Proof of the lemma used on page 5: We will prove that

$$\nabla_{\mathbf{p}} \mu / \mathbf{p} / = -\text{const.} \nabla_{\mathbf{p}} \nu / \mathbf{p} /.$$

If it is true, then if

$$\nabla_{\mathbf{p}} \mu / \mathbf{p} / = 0, \text{ then } \nabla_{\mathbf{p}} \nu / \mathbf{p} / = 0, \text{ and}$$

$$\text{sgn}(\nabla_{\mathbf{p}}^2 \mu / \mathbf{p} /) = -\text{sgn}(\nabla_{\mathbf{p}}^2 \nu / \mathbf{p} /).$$

$$\nabla_{\mathbf{p}} \mu / \mathbf{p} / = \nabla_{\mathbf{p}} \sum_{i=1}^N (\phi_i - \psi_i(\mathbf{p}))^2 =$$

$$= \sum_{i=1}^N -2(\phi_i - \psi_i(\mathbf{p})) \nabla_{\mathbf{p}} \psi_i(\mathbf{p}) =$$

$$= -2 \sum_{i=1}^N \phi_i \nabla_{\mathbf{p}} \psi_i(\mathbf{p}) - \sum_{i=1}^N 2\psi_i(\mathbf{p}) \nabla_{\mathbf{p}} \psi_i(\mathbf{p}) =$$

$$= -2 \nabla_{\mathbf{p}} \nu(\mathbf{p}) - \nabla_{\mathbf{p}} \left(\sum_{i=1}^N \psi_i^2(\mathbf{p}) \right) = -2 \nabla_{\mathbf{p}} \nu(\mathbf{p})$$

Q.e.d.

Appendix B

The proof of the theorem:

Let's use the notations of Fig. 8. We shall deal only with the case if $45^\circ \leq \alpha < 90^\circ$, because all other cases can be handled symmetrically. First we show that in the case of the line on Fig. 8, the statement of the theorem is valid. Here the quotient of the two integrals is:

$$\frac{\int_I f_2 f}{\int_I f_1 f} = \frac{T_2 + T_1}{T_2 - T_1}$$

Considering the sides of D to be one, we have

$$y = \text{tg} \beta = \text{tg}(45^\circ - \alpha) = \frac{\text{tg} - 1}{1 + \text{tg} \alpha}$$

and so

$$\frac{T_2 + T_1}{T_2 - T_1} = \frac{(1+y)m}{(1-y)m} = \frac{1+y}{1-y} = \text{tg} \alpha$$

Consequently, if e' lay above the line examined just now, then because of the similarity of all relevant triangles [see Fig. 9/ the statement would be true for e' as well:

$$T'_1 = k T_1, \quad T'_2 = k T_2,$$

$$\frac{T'_1 + T'_2}{T'_2 - T'_1} = \frac{kT_1 + kT_2}{kT_2 - kT_1} = \frac{T_1 + T_2}{T_2 - T_1} = \text{tg} \alpha.$$

Now if e' lay between e and the centre of D, then using the notations of Fig. 10 on account of the similarity of the triangles T_1 and T_2 UT₃UX here and T_1 and T_2 on Fig. 10 we have

$$\frac{T_2 + T + X + T_1}{T_2 + T_3 + X - T_1} = \text{tg} \alpha.$$

So we have to show only that

$$\begin{aligned} (\text{tg} \alpha) &= \frac{T_2 + T_3 + X + T_1}{T_2 + T_3 + X - T_1} = \\ &= \frac{T_1 + T_2 - T_3}{T_2 + T_3 - T_1} = \frac{\int_I f f_2}{\int_I f f_1} \end{aligned}$$

Subtracting 1 from both side, taking their reciprocal and adding 1 again we have

$$\frac{T_2 + T_3 + X}{T_1} = \frac{T_2}{T_1 - T_3} \quad \text{i. e.}$$

$$\frac{T_1}{T_2 + T_3 + X} = \frac{T_3}{T_3 + X}.$$

This means we have to show only that

$$\frac{DE}{EB} = \frac{FG}{FB}$$

And this is true for, because of the similarities of the triangles FJB and CAB, $\frac{FG}{FB} = \frac{CD}{CB}$; so it is enough to show that $\frac{DE}{EB} = \frac{CD}{CB}$, i. e. that $\frac{CD}{DE} = \frac{CB}{EB}$

Now for AD is the bisector of the angle CAE, we have:

$$\frac{CO}{DE} = \frac{CA}{AE}, \text{ and for the similarity of the}$$

triangles CAB and EHB

$$\frac{CB}{EB} = \frac{CA}{EH}$$

and for the triangle AEH is isocles $AE=EH$, and the theorem is proved in this case, too. If e were below the centre of D, the same proof could be presented for negative d . /We have nowhere made use of the positive property of D./ So the whole theorem is proved.

References

- (1) Deutsch, S.: Conjectures on mammalian neuron networks for visual pattern recognition. IEEE Trans. System Science and Cybernetics, SSC-2, pp. 81-85/1966/
- (2) Hawkins, J.K.: Photographic techniques for extracting image shapes. Photo-

graphic Science and Engineering, 8, pp. 329-335 /1964/

- (3) Holdermann, F.; Kazmierczak, H.: Pre-processing of grey-scale pictures. Computer Graphics and Image Processing, 1, pp. 66-80 /1972/
- (4) Hubel, D.H.; Wiesel, T.N.: Receptive fields, binocular interaction, and functional architecture of the cat's visual cortex. J. Physiol., 160, pp. 106-154 /1962/
- (5) Hueckel, M.H.: An operator which locates edges in digitized pictures. Journal of ACM, 18, pp. 113-125 /1971/
- (6) Hueckel, M.H.: A local visual operator which recognized edges and lines. Journal of ACM, 20, pp. 634-647 /1973/
- (7) Levine, M.D.: Feature extraction: a survey. Proc. IEEE, 57, pp. 1391-1407 /1969/
- (8) Martelli, A: Edge detection using heuristic search methods. Computer Graphics and Image Processing, 1, pp. 169-182 /1972/
- (9) Montanari, U.: On the optimal detection of curves in noisy pictures. Comm. ACM, 14, pp. 335-345 /1971/
- (10) McLeod, I.D.G.: Comments on "Techniques for edge detection". Proc. IEEE, 60, pp. 1344 /1972/
- (11) Nadler, M.: An analog-digital character recognition system. IEEE Trans. on Electronic Computers, EC-12, pp. 814-821 /1963/
- (12) Nilsson, N.J.: Adaptive pattern recognition: a survey. 1966 Bionics Symp., Dayton, 1966
- (13) Rosenfeld, A.: Picture processing by computer. Technical report 68-71, Univ. of Maryland, Computer Science Centre, College Park, Maryland /1968/
- (14) Rosenfeld, A.: A nonlinear edge detection technique. Proc. IEEE, 58, pp. 814-816 /1970/
- (15) Rosenfeld, A.; Thurston, M.: Edge and curve detection for visual scene analysis. IEEE Trans. on Computers, C-20, pp. 562-569 /1971/
- (16) Rosenfeld, A.; Thurston, M.; Lee, Y.: Edge and curve detection: further experiments. IEEE Trans. on Computers, C-21, pp. 677-715 /1972/
- (17) Shirai, Y.: A step toward context-sensitive recognition of irregular objects. Computer Graphics and Image Processing, 2, pp. 298-307 /1973/

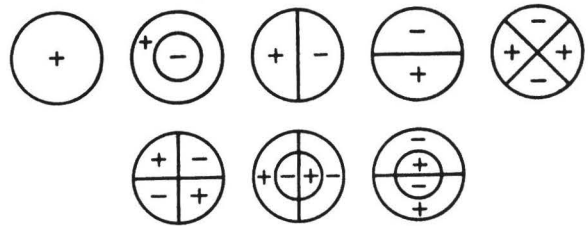


Fig 1



Fig 2

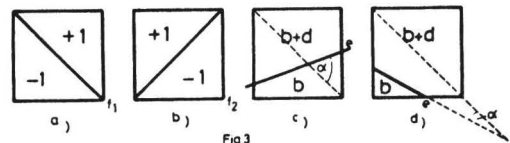


Fig 3



Fig 4

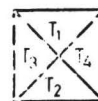


Fig 5



Fig 6

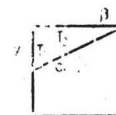
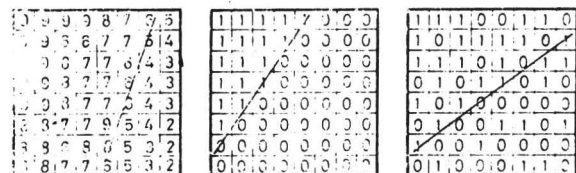


Fig 8

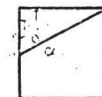


Fig 9

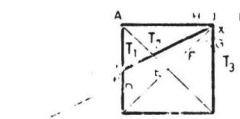


Fig 10