Journal of Electronic Imaging 6(4), 415–425 (October 1997).

Selecting appropriate difference operators for digital images by local feature detection

Peter Veelaert
University of Ghent
Department ELIS
B-9000 Ghent, Belgium
E-mail: veelaert@elis.rug.ac.be

Abstract. *An often reoccurring problem in digital image processing is the application of operators from differential geometry to discrete representations of curves and surfaces. We propose the use of feature detectors to improve the estimation of differentials of discrete functions. To this end we replace a differential operator by a bank of feature detectors and difference operators. The purpose of the feature detectors is first to examine the local behavior of the function. Next, depending on the outcome, the feature detectors select the most appropriate difference operator. For example, if the function behaves locally as a linear function, they select a difference operator that is well suited for linear functions. We show that this technique can be put on a firm mathematical basis. In particular, when designing a bank of feature detectors, we use Groebner bases for the functional decomposition and combination of the detectors. We illustrate the mathematical results with several practical examples.* © *1997 SPIE and IS&T.* [S1017-9909(97)00304-8]

1 Introduction

An often reoccurring problem in digital image processing is the application of operators from differential geometry to digital representations of curves and surfaces. To detect edges in an image we look for the relative extrema in the first directional derivative of the image function. To analyze a digital representation of a curve or surface, e.g., the representation of a heart obtained by computer tomography, we look for tangent planes, points of maximal curvature, surface normals, geodesics, shape operators, and Gauss maps. For continuous surfaces these are all standard functions that can be computed by well-defined differential operators.[1–3] Differential operators, however, cannot be applied directly to digitized surfaces or digitized curves. An obvious solution is to replace the differential operator by a discrete operator, i.e., a difference operator. For instance, in their work on edge detection, several authors proposed difference operators to mimic the operation of taking first or second order derivatives (see Ref. 4).

In this paper, we propose to replace a differential operator not by a single difference operator but by a combination of feature detectors and difference operators. The purpose of the feature detectors is to examine the local behavior of the function. Next, depending on their outcome, the feature detectors select the most appropriate difference operator. For example, if the function behaves locally as a linear function, they select a difference operator that is well suited for linear functions. If it behaves like a quadratic function, they select an operator well suited for quadratic functions. The replacement of a differential operator by a bank of difference operators involves three important choices: (1) the kind of features, i.e., whether we should look for linear, quadratic, symmetric, or other behavior; (2) the size of the neighborhood in which we are looking for a feature; and (3) for each detected feature, the most appropriate difference operator. One of the main goals of this paper is to introduce formal results that enable us to make these choices. We show that features can be sought for in a systematic way, and that for each feature and size of the neighborhood there is a best choice for the difference operator.

The most extensive work on the application of differential operations to digital images was done in edge detection and in image enhancement while preserving edges. As argued by Torre and Poggio differentiation of an image corrupted by noise is an ill-posed problem in the sense of Hadamard.[5] Therefore, assumptions must be made about the smoothness of the image, the distribution of the noise, and the most important criteria that an edge must satisfy. Under such assumptions, so-called optimal edge detectors were derived by Canny,[6] Sarkar and Boyer,[7] and more recently by Mehrotra and Zhan.[8]

One way to take into account the local smoothness of an image is by fitting a function to it, as was done by Haralick to estimate the second order derivatives of an image.[9] As pointed out by Nalwa and Binford, one of the major problems of differentiation by using fitting functions, is the choice of an appropriate basis of fitting functions.[10] To solve this problem they proposed to fit functions to the image that are not chosen from one but from several bases of fitting functions, and to accept the fitting function that is adequate in the least squares sense and that has the fewest parameters. Similarly, the choice of the neighborhood in which the fitting is done may greatly influence the performance of an edge detector. After a close examination of some of the defects of commonly used edge detectors,

Paper IST-03 received Jan. 7, 1997; revised manuscript received May 7, 1997; accepted for publication May 29, 1997. This paper is a revision of a paper presented at the SPIE conference on Vision Geometry V, Aug. 1996, Denver, CO. The paper presented there appears (unrefereed) in SPIE Proceedings Vol. 2826.
1017-9909/97/$10.00

Fleck proposed an edge detector that uses not one but multiple widths to compute the first order differences of an image.[11,12] Recently, the use of multiscale approaches and adaptive techniques for edge detection and image smoothing have become quite popular.[13,14]

The main points of difference or resemblance of our approach when compared with previous work are the following:

- We explicitly assume that the image is digital, and that the main source of noise is quantization noise (unlike many ''optimal'' edge detectors that often assume Gaussian noise, which may not always be very realistic for digital images).
- To detect features we use multiple bases of fitting functions.
- We consider fitting in neighborhoods of different sizes.
- In our approach, feature detection is based on uniform fitting, which seems to be the most appropriate form of fitting for digitized curves and surfaces.
- We use Groebner bases as a powerful mathematical technique to design the feature detectors.
- We derive precise error bounds for the estimation of the differentials.

The main contribution, however, is that we introduce a formal technique to replace a differential operator by a set of difference operators. This technique has been inspired by two different lines of research. On the one hand, a careful analysis of 2-D difference operators leads to the use of Groebner bases, which only relatively recently became available as a powerful technique to study the structure of polynomial ideals.[15] On the other hand, in digital geometry, digitized curves and surfaces can be characterized with inequalities instead of equations. To manipulate these inequalities we can use the method of legal linear dependences, which was introduced by Fourier and Motzkin.[16,17] This method generates all possible relations that follow from a given set of inequalities. It can tell us what information can be derived from the approximation of a continuous function by a discrete one, and in particular, it can tell us about the approximation of a differential by a difference.

The emphasis of this paper is on the theoretical results, although we also provide some experimental results to illustrate the approach. Since some of the mathematical tools may not be familiar to a computer vision oriented audience, we first introduce the general idea in Section 2. Next, in the sections that follow we gradually fill in more of the mathematical details.

2 Digitizing Differential Operators

We introduce the main ingredients that we will need to ''digitize'' a differential operator.

2.1 *Feature Detection*

In what follows f denotes a digitized function $f:\mathbf{Z}^m \rightarrow \mathbf{Z}$. A typical graph of such a function is shown in Figure 1. To determine the behavior of f at a point x_0, or in other words to determine its features, we shall consider the error of fit in a finite neighborhood $D \subset Z^m$ of x_0. To be precise, let G be a given class of fitting functions $\tilde{g}:\mathbf{R}^m \rightarrow \mathbf{R}$. We write $|f-\tilde{g}| \leq \epsilon$ as a shorthand for $|f(x)-\tilde{g}(x)| \leq \epsilon$ for all $x \in D$. We say that f behaves locally as a G-like function, if there is a fitting function $\tilde{g}$ in G such that $|f-\tilde{g}| \leq \epsilon$ for a given threshold ϵ. If we know that the only noise present in an image is quantization noise, then we can set $\epsilon = 1/2$.

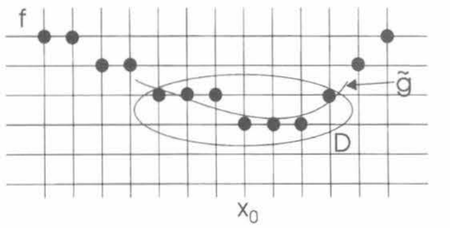


Fig. 1 Finding a tangent to a digitized curve.

2.2 *Difference Operators*

We express difference operators in terms of shift operators. The shift operator σ^i is defined by $\sigma^i \tilde{g}(x) = \tilde{g}(x+i)$, for $x,i \in Z^m$. The functional composition of shift operators can be expressed as a multiplication of polynomials, i.e., $\sigma^i \sigma^j \tilde{g} = \sigma^{i+j} f$. A difference operator P can be represented as a polynomial in σ, that is, $P = \Sigma_{i=0}^{l} p_i \sigma^i$. If we write that $P\tilde{g} = 0$, for some difference operator $P = \Sigma p_i \sigma^i$, then this means that $\Sigma p_i \sigma^i \tilde{g}(x) = 0$ for all $x \in \mathbf{Z}^m$. If we write that $|Pf - P\tilde{g}| \leq \epsilon$, this means that $|Pf(x) - P\tilde{g}(x)| \leq \epsilon$ for all x for which $Pf(x)$ is well defined, in other words, we must have $(x+i) \in D$ for every nonvanishing coefficient p_i of the difference operator P.

We often represent 2-D difference operators by templates. Suppose we have a 2-D difference operator $P = \Sigma p_i \sigma^i$, $i \in Z^2$, or written out fully, $P = \Sigma_{i_x, i_y} p_{i_x i_y} \times \sigma_x^{i_x} \sigma_y^{i_y}$. This difference operator can be represented by its 2-D template:

p_{00}	p_{10}	p_{20}
p_{01}	p_{11}	p_{21}
p_{02}	p_{12}	p_{22}
p_{03}	...	

We use the convention that the box at the upper left corner corresponds to p_{00}. Boxes with vanishing coefficients are either not drawn, or drawn as empty boxes.

2.3 *Decision Trees*

There is no unique way to replace a differential by a difference operator. For example, we could replace the differential d/dx by the operator $\Delta_x = \sigma_x - 1$, since at least for linear functions of the form $\tilde{g}(x) = ax+b$, both operators yield the same result, that is, $d\tilde{g}/dx = \Delta_x \tilde{g} = a$. However, also the difference operators $(\sigma^2-1)/2$, and $(\sigma^3-1)/3$ yield the first derivative when applied to a linear function.

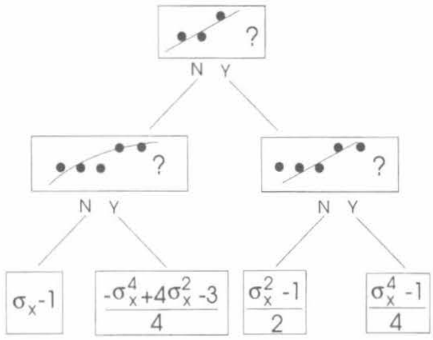


Fig. 2 Decision tree used to estimate a differential.

To complicate matters even more, for a quadratic function $\tilde{g}(x)=ax^2+bx+c$, the operator Δ_x does not yield the first derivative of $\tilde{g}$, instead we have, for example, $d\tilde{g}/dx = (\Delta_x - \Delta_x^2/2)\tilde{g} = 2ax+b$.

To remove the arbitrariness in the choice of an appropriate difference operator, we propose to use a bank of feature detectors and difference operators. This bank can be represented as decision tree, of which Figure 2 shows one example. The nonterminal nodes of the tree consist of feature detectors. At the top of the tree we first examine whether the function f is linear, or close to linear as defined by the threshold ϵ, in a three-point neighborhood of x_0. If this is the case, we tighten the constraints a little bit, and we verify whether the function is still linear in a five-point neighborhood of x_0. On the other hand, if the function is not linear at all, we verify whether it is parabolic in a five-point neighborhood. Next, depending on the outcome of the feature detection nodes in the tree, we select an appropriate difference operator. For example, if the function is linear in a five-point neighborhood, we choose the operator $(\sigma^4-1)/4$. If the function turns out to be parabolic rather than linear, we choose the operator $(-\sigma_x^4+4\sigma_x^2-3)/4$. If no interesting features are detected at all, we choose the simplest possible operator $\sigma_x - 1$ as a last resort. Note that to simplify the notation of the difference operators, we often choose a neighborhood that extends to the right side of x_0, e.g., the three-point neighborhood $\{x_0, x_0+1, x_0+2\}$.

The decision tree captures the main idea of our approach. In the remainder of the paper, we gradually fill in more details about the mathematical structure, contents and possible extensions of the tree.

3 More Structure for the Decision Tree

In this section, we explain why there is a decision tree in the first place, and why the detection of features may help us to choose the most appropriate difference operator. Our first goal is to establish a simple connection between the fitting error for f and the error that results when we replace a differential by a difference operator.

3.1 *Replacing Differentials by Difference Operators*

Let L be the differential operator that must be replaced by a difference operator. As mentioned previously, we can find such a replacement by selecting a class of fitting functions G and a difference operator Q that works well for this class. To be precise, we have a difference operator $Q = \Sigma q_i \sigma^i$ such that $Q\tilde{g} = L\tilde{g}$, for every $\tilde{g}$ in the class G.

Theorem 1. Let $\tilde{g}$ be an approximation for f such that $|f-\tilde{g}| \leq \epsilon$. Then we have $|Qf - L\tilde{g}| \leq \epsilon \Sigma |q_i|$.

Proof. Let $|f(x)-\tilde{g}(x)| \leq \epsilon$, for x in $\mathbf{Z}^m$. Then, clearly $|q_i\sigma^i f(x) - q_i \sigma^i \tilde{g}(x)| \leq \epsilon |q_i|$, for x,i in $\mathbf{Z}^m$. If we add all the left sides and all the right sides of the preceding inequality for all terms $q_i\sigma^i$ of the difference operator Q, then we find $|Qf(x)-Q\tilde{g}(x)| = |Qf(x) - L\tilde{g}(x)| \leq \epsilon\Sigma|q_i|$.

Hence, the difference operator Q will be a good approximation for the differential operator L provided the class of fitting functions contains at least one function $\tilde{g}$ that is a good approximation for f.

3.2 *More Structure for the Feature Detectors*

Theorem 1 still contains the uncertainty that the class of fitting functions may or may not contain a good approximation for the digitized function f. We now try to eliminate the explicit occurrence of a fitting function in Theorem 1 by imposing additional constraints on the class of fitting functions. To this end we demand that the fitting functions $\tilde{g}$ satisfy a (possibly infinite) system of difference equations, i.e., $P_i\tilde{g}=0$, or in other words $G=\{\tilde{g} \mid P_i\tilde{g}=0, \text{ for } i=1,2,\ldots\}$. We use the operators P_i to eliminate $\tilde{g}$ from the inequality $|f-\tilde{g}|\leq\epsilon$. However, the set of operators that can be used to eliminate the fitting function is much larger. We also have $P_1P_2\tilde{g}=0$, $(P_1+P_2)\tilde{g}=0$, and in fact, for arbitrary difference operators $Q_i \in \mathbf{R}[\sigma]$, we have $(\Sigma Q_i P_i)\tilde{g}=0$.

The set of all possible difference equations satisfied by $\tilde{g}$ is best described as the polynomial ideal I generated by the set of difference operators P_i. Such an ideal consists of all difference operators of the form $\Sigma A_i P_i$, where the A_i are arbitrary polynomials in σ. We write $I=\langle P_1, P_2, \ldots\rangle$, and we say that the polynomials P_i form a basis for the ideal I. Thus, $P\tilde{g}=0$ for any operator P in the ideal $\langle P_1, P_2, \ldots \rangle$.

The introduction of the difference equations $P_i\tilde{g}=0$ has serious implications with regard to the use of fitting functions. First, it follows that the class of fitting functions will be shift invariant, that is, if $\tilde{g}$ is a fitting function then $\sigma^i\tilde{g}$ is also a fitting function. As a second consequence, by eliminating the explicit occurrence of $\tilde{g}$, we can also eliminate the computation of a fitting function for f.

Theorem 2. There is a fitting function $\tilde{g}\in G$ such that $|f-\tilde{g}|\leq\epsilon$ if and only if $|Pf|\leq\epsilon\Sigma|p_i|$ for every $P\in I$.

Proof. Assume that f satisfies the inequality $|Pf| \leq \epsilon\Sigma|p_i|$ for every polynomial in I. Then we must prove that the system

$$\begin{cases} |\tilde{g}(x)-f(x)| \leq \epsilon \\ \sum p_i \tilde{g}(x+i) = 0, \end{cases} \tag{1}$$

where $x\in D$ and $\Sigma p_i\sigma^i \in \{P_1,\ldots,P_n\}$, has a solution for

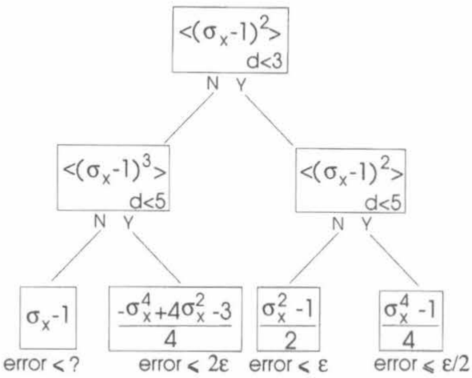


Fig. 3 More details for the decision tree.

the indeterminates $\tilde{g}(x)$. This is a system of linear inequalities that, according to the Kuhn-Fourier theorem (see Stoer and Witzgall[16]), is solvable if and only if each of its legal linear dependences leads to a zero relation that is always true. In fact, for this particular system, all legal linear dependences lead to zero relations of the form $|Pf| \leqslant \epsilon\Sigma|p_i|$, where $P=\Sigma p_i\sigma_i$ is a polynomial of I, and conversely, every polynomial P of I leads to a relation of the above form. By assumption, these relations are true.

As suggested by Theorem 2, from now on we can verify whether f has the right features without bothering about which fitting function would actually yield the closest fit. The following is an immediate corollary of the previous theorems. It involves all the elements on which our proposition is based: the combined use of feature detectors and a difference operators.

Corollary 1. Let P_i, where $i=1,\ldots,n$, be a finite set of difference operators. Let f be a function satisfying $|Pf| \leqslant \epsilon\Sigma|p_i|$ for all operators P in the operator ideal $I = \langle P_1,\ldots,P_n\rangle$. Let Q be an arbitrary difference operator. Then there exists a function $\tilde{g}$ satisfying $P_i\tilde{g}=0$ such that

$$|Qf-L\tilde{g}| \leqslant \epsilon \sum |q_j|.$$

With regard to the uncertainty as to applying an appropriate difference operator, this corollary is an important improvement over Theorem 1. It is now sufficient to verify whether f has the right features so that we can apply the appropriate difference operator Q. To this end we must verify whether f satisfies $|Pf| \leqslant \epsilon\Sigma|p_i|$, for P in I, since in that case, there is a good approximation for which the operator Q gives the correct result. The operator Q and the feature verifying operators P are linked to each other by the fitting functions $\tilde{g}$. The fitting functions satisfy $P\tilde{g}=0$, and Q must be chosen such that $Q\tilde{g}=L\tilde{g}$.

We have now already filled in some of the details for the feature detection nodes of the decision tree, which is illustrated in Figure 3. For example, the feature detection node at the root of the tree refers to all polynomials P in the ideal $\langle(\sigma_x-1)^2\rangle$ of degree d less than 3. If the function f satisfies $|Pf| \leqslant \epsilon\Sigma|p_i|$ for each polynomial in this ideal, then f is linear or close to linear in a three-point neighborhood of x_0, and as a result we can use a difference operator appropriate for linear functions.

3.3 *More Structure for the Difference Operators*

Adding structure to the feature detectors has direct consequences for the contents of the terminal nodes of the decision tree where we select a difference operator. In fact, once the class of fitting functions G has been chosen, it remains to choose an appropriate difference operator Q that satisfies $Q\tilde{g}=L\tilde{g}$. Since $(Q+P)\tilde{g}=Q\tilde{g}$ for any P in I, there seem to be many possible ways to choose Q. The following lemma states, however, that there are no other possibilities than those provided by the ideal I. A proof is given in the Appendix.

Lemma 1. Let $\tilde{g}:R^m \to R$ be an arbitrary real function and let I be the ideal of operators P for which $P\tilde{g}=0$. Let Q be an arbitrary difference operator. Then any operator R satisfying $R\tilde{g}=Q\tilde{g}$ can be written as $R=Q+P$, where $P \in I$.

This clearly specifies which difference operators are valid candidates to estimate $L\tilde{g}$. In fact, we can look for the best possible candidate. For any operator P in I we have

$$|(P+Q)f-L\tilde{g}| \leqslant \epsilon \sum |p_j+q_j|.$$

If we look in I for a difference operator P for which the right side of the preceding inequality becomes minimal, then $P+Q$ will give the lowest error when used to approximate the differential L. In principle, since the polynomial ideal I also has the structure of a real vector space, the best operator can be found by solving a linear programming problem, provided we eliminate the absolute values by taking into account the signs of p_j+q_j.

We illustrate Lemma 1 for the decision tree of Figure 3. We must replace the differential operator d/dx by a difference operator. Part of the feature detection process involves linear fitting functions of the form $\tilde{g}(x)=ax+b$. The simplest idea is to replace d/dx by the difference operator (σ_x-1). However, if we know that f behaves like a linear function in a large neighborhood of x_0, we can obtain a better result. Here $I=\langle(\sigma_x-1)^2\rangle$ is the ideal of templates P for which $P\tilde{g}=0$. According to Corollary 1, for any operator of the form

$$Q=(\sigma_x-1)+R(\sigma_x-1)^2,$$

where R is an arbitrary difference operator, the following inequality

$$|Qf-\mathrm{d}\tilde{g}/\mathrm{d}x| \leqslant \epsilon \sum |q_j| \tag{2}$$

is satisfied. In particular, we have

$$\left|\frac{\sigma_x^k-1}{k}f-\mathrm{d}\tilde{g}/\mathrm{d}x\right| \leqslant 2\epsilon/k,$$

for k any positive integer. If the approximation of f by a linear fitting function is excellent in a large neighborhood, then we can choose k large to obtain a low value for the error bound $2\epsilon/k$. As shown in Figure 3, if f is close to linear in a five-point neighborhood, we can choose $(\sigma_x^4 - 1)/4$.

Second, at the left side of the tree we use fitting functions of the form $\tilde{g}(x)=ax^2+bx+c$ to detect quadratic behavior. In this case, we have $I=\langle(\sigma_x-1)^3\rangle$, or in other words, any difference operator P for which $P\tilde{g}=0$ is a multiple of the third order difference operator $\Delta_x^3=(\sigma_x-1)^3$. The inequality of Eq. (2) now holds for every operator of the form

$$Q=(\sigma_x-1)-(\sigma_x-1)^2/2+R(\sigma_x-1)^3.$$

In particular, we have a first operator

$$|[(\sigma_x-1)-(\sigma_x-1)^2/2]f-\mathrm{d}\tilde{g}/\mathrm{d}x|\leq 4\epsilon,$$

but also a second operator

$$\left|\frac{(\sigma_x^4-1)-2(\sigma_x^2-1)^2}{4}f-\mathrm{d}\tilde{g}/\mathrm{d}x\right|\leq 2\epsilon.$$

Again, if we have a good parabolic fit in a five-point neighborhood, the second difference operator yields better results. Also note in Figure 3 that the difference operators for linear functions give better results than the difference operators for quadratic functions, since they do not have to compensate for the quadratic term ax^2. In other words, if the digitized function behaves locally as a linear function, it is appropriate to use difference operators derived for linear fitting functions. If the digitized function locally has a quadratic behavior, then we must fall back on difference operators derived for the quadratic functions, which perform slightly less.

With the preceding results we have now filled in some more details about the contents and structure of the decision tree, as illustrated in Figure 3. In particular, we have filled in the structure of the feature detectors and the most appropriate difference operators. Furthermore, for each difference operator, we have filled in the error bound. For example, the operator $(\sigma_x^4-1)/4$ has an error bound of $\epsilon/2$, which must be interpreted as follows: If f is close to linear in a five-point neighborhood of x_0, then if we apply this operator to f, we find a result that differs not more than $\epsilon/2$ from the first derivative of a linear function $\tilde{g}$ that is close to f, that is, $|f-\tilde{g}|\leq\epsilon$.

Note, however, that in its current form the decision tree cannot yet be used in a practical way. According to what we have up to now, to detect whether a function has a certain feature (linear or parabolic) we must verify whether $|Pf|\leq\epsilon\Sigma|p_i|$ for all polynomials P of degree less than d in a polynomial ideal I, of which there are an infinite number. The reduction to a finite number of polynomials is the subject of the next section.

4 Polynomial Bases for Feature Detectors

The difference operators P_i were introduced to define a class of fitting functions to improve our knowledge about the quality of the digitization of the differential operator L. We now take a closer look at the structure of the ideal $I=\langle P_1,P_2,\ldots\rangle$. First we note that a polynomial ideal always has a finite basis.

Theorem 3. Let $I=\langle P_1,P_2,\ldots\rangle$ be an ideal of difference operators generated by a possibly infinite set of operators P_i. Then I is finitely generated.

Proof. This is a restatement of Hilbert's basis theorem for polynomial ideals,[15] which states that any ideal of polynomials in the ring $\mathbf{R}[\sigma_x,\sigma_y,\ldots]$ can always be generated by a finite basis of polynomials.

Hence any ideal of polynomials in the ring $\mathbf{R}[\sigma_x,\sigma_y,\ldots]$ can always be generated by a finite basis of operators $\langle B_1,\ldots,B_n\rangle$. Or equivalently, even if a system has infinitely many difference equations, all these equations can be obtained by multiplying, adding and translating a finite set of basis equations.

Moreover, if we impose an ordering on the shift operators $\sigma_x,\sigma_y,\ldots$, then we can always find a so-called *Groebner basis* for the ideal I (Ref. 15). A basis of an ideal is called a Groebner basis for I if the leading term of any element of I is divisible by one of the leading terms of the polynomials of the basis. For example, if the polynomials are ordered according to lexicographic order, with $\sigma_x>\sigma_y$, then $\langle(\sigma_x-1)^2,(\sigma_x-1)(\sigma_y-1),(\sigma_y-1)^2\rangle$ forms a Groebner basis for a given polynomial ideal I. The leading term of any polynomial in the ideal is always divisible by one of the leading terms $\sigma_x^2,\sigma_x\sigma_y,\sigma_y^2$ of the basis. There exist efficient algorithms to calculate Groebner bases, and in turn these bases lead to efficient algorithms that can determine whether a given polynomial belongs to an ideal I (Ref. 15). In this paper, we make only limited use of Groebner bases, in particular to find a minimal system of difference equations for a given class of fitting functions.

Our next goal now is to apply Theorem 2. According to this theorem if we want to know whether there exists a good continuous approximation $\tilde{g}$ for f, we should verify whether $|Pf|\leq\epsilon\Sigma|p_i|$, for every difference operator in I. In some cases, i.e., when the solution space of the difference equations is a finite linear vector space, we can show that the feature can be detected without error by verifying only a finite number of inequalities. Assume therefore that the general solution of the partial difference equations $P_1g=0,\ldots,P_ng=0$ can be written as

$$\alpha_1g_1+\cdots+\alpha_lg_l.$$

That is, the solution set of the difference equations is a linear vector space that has $g_1,\ldots,g_l$ as a basis. Let K_D be the set of all difference operators of the form

$$\begin{vmatrix} g_1(x_1) & \cdots & g_l(x_1) & \sigma^{x_1} \\ \cdots & & & \\ g_1(x_{l+1}) & \cdots & g_l(x_{l+1}) & \sigma^{x_{l+1}} \end{vmatrix},$$

with $x_i \in D$. Thus the operators of K_D can be written as determinantal expressions of the coefficients $g_i(x_i)$ and the shift operators σ^{x_i}. Furthermore, let I_D denote the set of all the difference operators in I for which $Pf(x)$ is well defined for at least one x in D, that is $(x+i) \in D$ for every nonvanishing coefficient p_i of the difference operator P. The following theorem follows from an elimination theory for linear inequalities.[18] An outline of its proof is given in the Appendix.

Theorem 4. The polynomials of K_D form a finite basis for the ideal generated by I_D. Furthermore, if the function f satisfies the inequality $|Pf| \leq \epsilon \Sigma_i |p_i|$ for every polynomial P in K_D, then f will satisfy this inequality for all polynomials of I_D.

Note that this theorem imposes a constraint on the size of the neighborhood D in which we use the fitting function, that is D must be large enough such that I_D is equal to I. Or equivalently, I_D must contain a basis for I.

We illustrate these results for the decision tree of Figure 3. At the left side of the tree we use a feature detector for fitting functions of the form $\tilde{g}(x) = \alpha_1 x^2 + \alpha_2 x + \alpha_3$. In this case, the polynomials of the set K_D have the form

$$\begin{vmatrix} 1 & x_1 & x_1^2 & \sigma_x^{x_1} \\ \cdots & & & \\ 1 & x_4 & x_4^2 & \sigma_x^{x_4} \end{vmatrix}.$$

For example, for $\{x_1,\ldots,x_4\} = \{x_0, x_0+1, x_0+2, x_0+4\}$, the preceding determinant is equal to $\sigma_x^{x_0}(6 - 16\sigma_x + 12\sigma_x^2 - 2\sigma_x^4)$. After evaluating the determinants for all the four-point subsets of the five-point neighborhood $\{x_0, x_0+1, x_0+2, x_0+3, x_0+4\}$, we find that f behaves like a parabolic function provided f satisfies

$$|f(x_0) - 3f(x_0+1) + 3f(x_0+2) - f(x_0+3)| \leq 8\epsilon,$$

$$|3f(x_0) - 8f(x_0+1) + 6f(x_0+2) - f(x_0+4)| \leq 18\epsilon,$$

$$|f(x_0) - 2f(x_0+1) + 2f(x_0+3) - f(x_0+4)| \leq 6\epsilon,$$

$$|f(x_0) - 6f(x_0+2) + 8f(x_0+3) - 3f(x_0+4)| \leq 18\epsilon,$$

$$|f(x_0+1) - 3f(x_0+2) + 3f(x_0+3) - f(x_0+4)| \leq 8\epsilon.$$

Likewise, at the root of the tree, we find that f behaves like a linear function provided f satisfies $|f(x_0) - 2f(x_0+1) + f(x_0+2)| \leq 4\epsilon$.

From now on we have a practical implementation of the decision tree. Each feature detector can be implemented as a finite set of inequalities that must be verified. For example, to detect whether a function has a quadratic behavior in a five-point neighborhood, we must verify the preceding five inequalities.

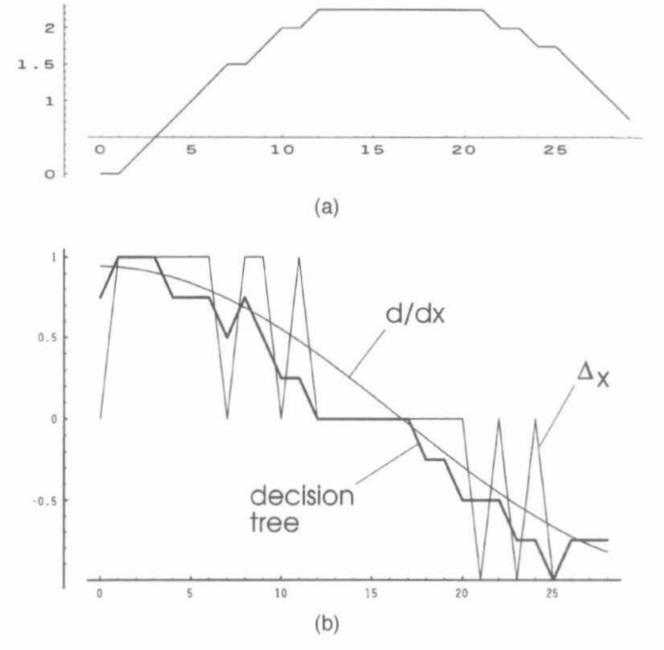


Fig. 4 Result of applying a decision tree to a sine wave.

4.1 *Some Experimental Results for a First Order Differential*

We briefly illustrate the application of the simple decision tree that we have designed in the previous sections. Figure 4(a) shows a digitized sine wave. Figure 4(b) shows a cosine as the first order differential d/dx of a sine wave. Furthermore, it shows the result of applying the difference operator Δ_x to the digitized sine wave, and the result of applying the decision tree of Figure 3 to the digitized sine wave, which is closer to the exact result.

Figure 5 shows the result of applying a decision tree that is the same as that of Figure 3 except for the fact that the neighborhoods were chosen to be symmetrical around

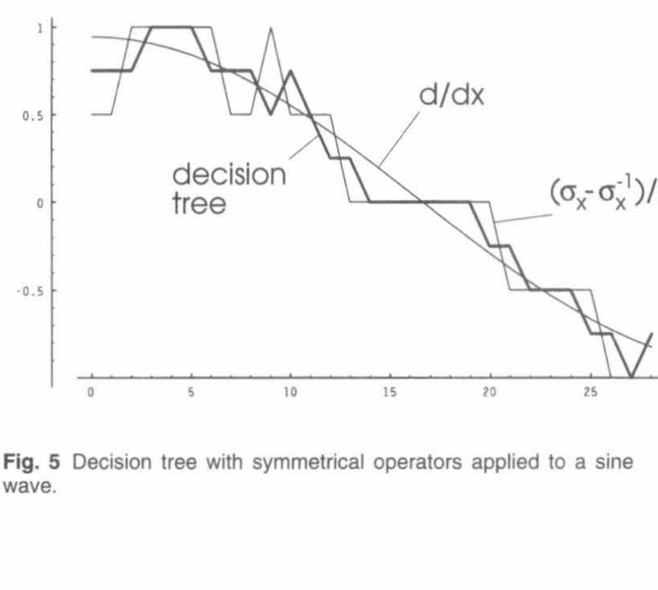


Fig. 5 Decision tree with symmetrical operators applied to a sine wave.

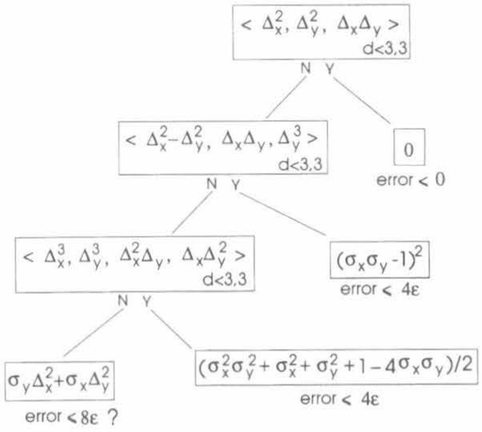


Fig. 6 Decision tree for the Laplacian.

x_0, e.g., as a five-point neighborhood we use $\{x_0-2, x_0-1, x_0, x_0+1, x_0+2\}$. Also shown is the result of applying the symmetrical difference operator $(\sigma_x-\sigma_x^{-1})/2$ to the data. From all the difference operators in the decision tree, this difference operator is by far the most suited for this particular sine wave, and therefore it was also selected most of the times by the feature detectors in the decision tree.

5 Two-Dimensional Operators

In this section, we discuss some of the peculiarities that arise for 2-D operators. This further illustrates the usefulness of Groebner bases. In particular, we design a decision tree for the Laplacian operator $\partial^2/\partial x^2+\partial^2/\partial y^2$, as shown in Figure 6. We use features derived from three different kinds of fitting functions:

$$\alpha_1 x+\alpha_2 y+\alpha_3,$$

$$\alpha_1(x^2+y^2)+\alpha_2 x+\alpha_3 y+\alpha_4,$$

$$\alpha_1 x^2+\alpha_2 y^2+\alpha_3 xy+\alpha_4 x+\alpha_5 y+\alpha_6.$$

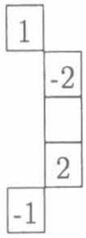
For all features, the chosen neighborhood is a 3×3 array centered around (x_0, y_0).

5.1 *Template Bases in Two Dimensions*

First we must find template bases for the feature detection nodes of the tree. We illustrate this for the fitting functions of the form $f(x,y)=\alpha_1(x^2+y^2)+\alpha_2 x+\alpha_3 y+\alpha_4$. As explained in a previous section, the polynomials of the set K_D have the form

$$\begin{vmatrix} 1 & x_1 & y_1 & x_1^2+y_1^2 & \sigma_x^{x_1}\sigma_y^{y_1} \\ \cdots & & & & \\ 1 & x_5 & y_5 & x_5^2+y_5^2 & \sigma_x^{x_5}\sigma_y^{y_5} \end{vmatrix}.$$

For example, for $\{(x_1,y_1),\ldots\}=\{(0,0),(1,1),(0,2),(1,3),(0,4)\}$, this determinant is equal to $P_0=-\sigma_y^4-2\sigma_x\sigma_y+2\sigma_x\sigma_y^3+1$, which corresponds to the template

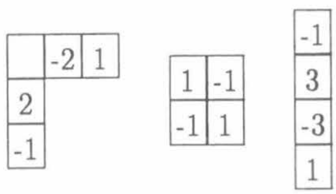


Thus by choosing different five-point subsets we can generate in a systematic way the polynomials of K_D that, according to Theorem 4, form a basis of I_D. Or, equivalently, we can generate templates that can recognize functions of the required form. In addition, we can find a Groebner basis for the ideal that describes all possible templates related to this feature. In general, a small sample of polynomials K_D suffices to calculate such a Groebner basis. In this case, we have generated a few subsets at random to find $\langle(\sigma_x-1)^2-(\sigma_y-1)^2, (\sigma_x-1)(\sigma_y-1), (\sigma_y-1)^3\rangle$ as a basis. In other words, for quadratic functions with circular symmetry, any difference operator P for which $P\tilde{g}=0$, can be written in the form

$$P=R_1(\Delta_x^2+\Delta_y^2)+R_2\Delta_x\Delta_y+R_3\Delta_y^3,$$

where the R_i are arbitrary difference operators. Thus we have found a concise description of all possible difference equations satisfied by quadratic circular symmetric functions, which is our first step in finding an appropriate discrete replacement for the Laplacian.

The Groebner basis that we have found corresponds to the templates



Because of the fundamental property of Groebner bases, every template of the feature detector can be written as a formal combination of the above three templates. More precisely, by using a simple division algorithm, we can decompose any template of I_D into combinations of the Groebner templates.[15] For example, the polynomial $P_0=-\sigma_y^4-2\sigma_x\sigma_y+2\sigma_x\sigma_y^3+1$ has the following decomposition:

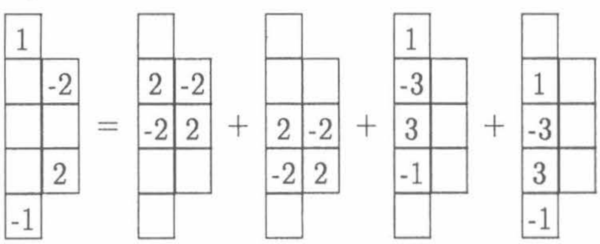
$$P_0=2\sigma_y(\sigma_y+1)(\sigma_y-1)(\sigma_x-1)-(\sigma_y+1)(\sigma_y-1)^3.$$

This corresponds to the following decomposition of the

Table 1 Functions and corresponding Groebner bases with lexicographic ordering where $\sigma_x > \sigma_y$.

Function	Groebner Basis
α_1	$\langle \Delta_x, \Delta_y \rangle$
$\alpha_1 x + \alpha_2$	$\langle \Delta_x^2, \Delta_y \rangle$
$\alpha_1(x+y) + \alpha_2$	$\langle \Delta_x - \Delta_y, \Delta_y^2 \rangle$
$\alpha_1 x + \alpha_2 y + \alpha_3$	$\langle \Delta_x^2, \Delta_x\Delta_y, \Delta_y^2 \rangle$
$\alpha_1 xy + \alpha_2 x + \alpha_3 y + \alpha_4$	$\langle \Delta_x^2, \Delta_y^2 \rangle$
$\alpha_1(x+y)^2 + \alpha_2(x+y) + \alpha_3$	$\langle \Delta_x - \Delta_y, \Delta_y^3 \rangle$
$\alpha_1(x^2+y^2) + \alpha_2 x + \alpha_3 y + \alpha_4$	$\langle \Delta_x^2 - \Delta_y^2, \Delta_x\Delta_y, \Delta_y^3 \rangle$
$\alpha_1(x+y)^2 + \alpha_2 x + \alpha_3 y + \alpha_4$	$\langle \Delta_x^2 - \Delta_y^2, \Delta_y(\Delta_x - \Delta_y), \Delta_y^3 \rangle$
$\alpha_1 x^2 + \alpha_2 y^2 + \alpha_3 x + \alpha_4 y + \alpha_5$	$\langle \Delta_x^3, \Delta_x\Delta_y, \Delta_y^3 \rangle$
$\alpha_1 x^2 + \alpha_2 y^2 + \alpha_3 xy + \alpha_4 x + \alpha_5 y + \alpha_6$	$\langle \Delta_x^3, \Delta_x^2\Delta_y, \Delta_x\Delta_y^2, \Delta_y^3 \rangle$

templates:



where the templates at the right side are combinations of the second and third templates of the Groebner basis.

Similarly, we can find the Groebner bases for other classes of fitting function. Table 1 lists some classes of fitting functions and their corresponding bases. For each class, the corresponding Groebner base completely characterizes the templates that will recognize the feature derived of the fitting functions.

Hence, we can now fill in bases for the ideals for the feature detecting nodes, which is shown in Figure 6. Figure 7 shows the same decision tree, but now with templates instead of polynomials.

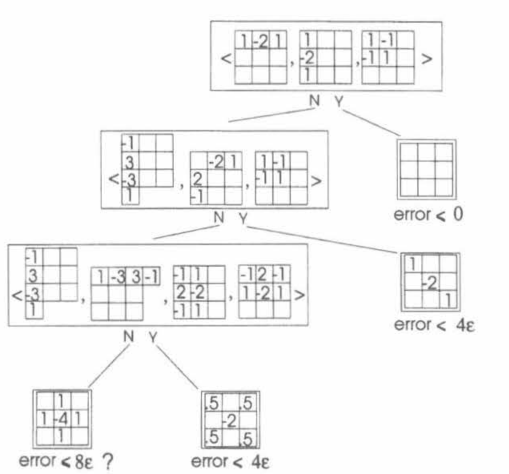


Fig. 7 Templates of the decision tree for the Laplacian.

5.2 *Difference Operators in Two Dimensions*

As before, once we have found the template bases for the polynomial ideals, we can look for the best difference operators. For example, for quadratic functions of the form

$$\tilde{g}(x,y) = \alpha_1 x^2 + \alpha_2 y^2 + \alpha_3 xy + \alpha_4 x + \alpha_5 y + \alpha_6,$$

we find that

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)\tilde{g} = (\Delta_x^2 + \Delta_y^2)\tilde{g} = 2\alpha_1 + 2\alpha_2.$$

Therefore, for quadratic functions any differential operator $Q(\sigma_x, \sigma_y)$ of the form

$$Q(\sigma_x, \sigma_y) = \Delta_x^2 + \Delta_y^2 + \Delta_x^3 R_1 + \Delta_y^3 R_2 + \Delta_x^2 \Delta_y R_3 + \Delta_x \Delta_y^2 R_4,$$

where the R_i are arbitrary polynomials in the shift operators, yields the exact value for the Laplacian. In particular, the operator $(\sigma_x^2\sigma_y^2 + \sigma_x^2 + \sigma_y^2 + 1 - 4\sigma_x\sigma_y)/2$ is a symmetric operator that has the preceding form, and for which $\Sigma|q_i|$ is as small as small possible. In Figures 6 and 7, this is the difference operator that has been filled in for functions that behave quadratically in a 3×3 neighborhood. Similarly, we have filled in the difference operators for the other features.

The difference operators obtained in this way lead to some interesting conclusions. Without feature detection we would probably choose the classical discrete equivalent of the Laplacian whose template is shown at the lower left side of Figure 7. However, when we find that the digitized function is locally linear or close to linear, the best possible difference operator is the zero operator, which yields an error equal to zero. In fact, the Laplacian of a linear function vanishes, and this is known without uncertainty. For quadratic functions with circular symmetry (the next level in the decision tree), we find that it is sufficient to compute the second order difference in a diagonal direction. Finally, for quadratic functions we find that the best difference operator has a template that is equal to the classical discrete Laplacian operator rotated over 45 deg and divided by 2.

6 Incomplete Feature Detection

Up to now we have explained the general method of digitizing differential operators. The selection of an appropriate difference operator depends on the detection of features. According to Theorem 4, a feature can be detected by verifying a finite set of inequalities. However, from a computational viewpoint this set may still be too large. However, Theorem 4 also has important consequences regarding the error characterizing parameters of smaller template bases. Suppose that instead of using the polynomials K_D we use only a small subset K'_D. As expected, this may lead to a possible misclassification of features. In terms of the decision tree, it means that it will increase the error of the estimation of the differential. We show that this error can actually be measured by examining how the polynomials K_D can be decomposed into polynomials of K'_D.

We assume that K'_D contains a minimum number of necessary polynomials. We assume that K'_D contains at least a basis for the ideal I, and that if $P_i \in K'_D$ and $\sigma^j P \in K_D$, then also $P \in K'_D$. Let $P_1,\ldots,P_n$ be a basis of K'_D. Thus, the polynomials P_i generate the ideal I, which, as we have seen, has a subset K_D of polynomials corresponding to templates that can detect the feature without errors. Since the polynomials P_i form a basis of the ideal I, every polynomial P of K_D can be written as

$$P = A_1 P_1 + \cdots + A_n P_n .$$

Now we make the important assumption that there is a decomposition such that every term $A_i P_i$ of this decomposition belongs to I_D. With this assumption the value of $|A_i P_i f|$ is well defined, and it follows that

$$|Pf| \leqslant |A_1 P_1 f| + \cdots + |A_n P_n f|. \tag{3}$$

As a result the values of $|A_i P_i f|$ impose an upper bound on the value of $|Pf|$. In particular, if f satisfies the inequalities corresponding to the polynomials of K'_D, then we have $|A_i P_i f| \leqslant \epsilon(\Sigma_j |a^i_j|)(\Sigma_j |p^i_j|)$. The right side of this inequality is a fixed number depending on the coefficients of P_i and P, but independent of f.

This kind of error analysis can be done for every polynomial P of K_D. Consequently, there is a global parameter $r \geqslant 0$ such that if f satisfies the inequalities of the original templates, then f also satisfies $|Pf| \leqslant (r+1)\epsilon \Sigma |p_j|$ for every P in K_D.

Hence, in general, to detect whether there is good approximation for the function f, we do not have to verify all the conditions $|Pf| \leqslant \epsilon \Sigma |p_i|$ for all polynomials P in the ideal K_D. It suffices to use a small but well chosen set of polynomials $P_1,\ldots,P_n$, for which the error parameter r is small. If the digitized function f satisfies the preceding inequality for the templates P_i, then there will be a fitting function $\bar{g}$ such that $|f - \bar{g}| \leqslant (r+1)\epsilon$.

6.1 *Incomplete Feature Detection in Two Dimensions*

Also in the decision tree for the Laplacian, we can reduce the number of inequalities that must be verified to detect a feature. As explained before, it is sufficient to verify only a small number of inequalities if we accept an additional increase of the maximal error by a factor r. For example, suppose that to find out whether f behaves like a quadratic function with circular symmetry, we verify only whether f satisfies the inequalities of the templates corresponding to the Groebner basis (at the second level of the tree):

$$|\sigma^i_x \sigma^j_y [(\sigma_x - 1)^2 - (\sigma_y - 1)^2] f| \leqslant 6\epsilon,$$

$$|\sigma^i_x \sigma^j_y (\sigma_x - 1)(\sigma_y - 1) f| \leqslant 4\epsilon,$$

$$|\sigma^i_x \sigma^j_y (\sigma_y - 1)^3 f| \leqslant 4\epsilon.$$

How will this increase the maximal error? From the previous decomposition of the polynomial $P_0 \in K_D$ it follows that

$$|P_0 f| \leqslant 4|(\sigma_x - 1)(\sigma_y - 1)f| + 2|(\sigma_y - 1)^3 f|.$$

This implies that $|P_0 f| \leqslant (4\times 4 + 2\times 4)\epsilon = 24\epsilon$. Hence, although f may not satisfy the inequality $|P_0 f| \leqslant 6\epsilon$, which is necessary to guarantee errorless detection, it is ensured that it satisfies the weaker inequality $|P_0 f| \leqslant 24\epsilon$. Hence the maximal error will increase by at least a factor 4.

6.2 *Decompositions Inside Rectangular Neighborhoods*

We mention one final peculiarity that arises when we decompose 2-D templates. In Eq. (3) we assume that we are using a template basis $P_1,\ldots,P_n$ such that every polynomial P of K_D can be decomposed as $P = A_1 P_1 + \cdots + A_n P_n$, where all terms $A_i P_i$ belong to I_D. The polynomials of a Groebner basis, and the determinantal expressions of Theorem 2 often provide a useful starting point to find such template bases.

We illustrate this for the detection of linear functions. We use the templates

$$\begin{array}{|c|c|c|}\hline 1 & -2 & 1 \\ \hline\end{array} \qquad \begin{array}{|c|c|}\hline 1 & -1 \\ \hline -1 & 1 \\ \hline\end{array} \qquad \begin{array}{|c|}\hline 1 \\ \hline -2 \\ \hline 1 \\ \hline\end{array} \tag{4}$$

to detect whether f is a linear function of x and y. Let the region D be an $M\times N$ rectangle. We denote K_D as K_{MN} and I_D as I_{MN}. The corresponding polynomials $P_1 = (\sigma_x - 1)^2$, $P_2 = (\sigma_x - 1)(\sigma_y - 1)$, and $P_3 = (\sigma_y - 1)^2$ generate an ideal I, and in fact, form a Groebner basis for I. Clearly, every polynomial P of K_{MN} has a decomposition of the form $P = A_1 P_1 + A_3 P_3 + R$, where none of the terms $A_i P_i$ has degree higher than M in σ_x or higher than N in σ_y, and R has the form $R = a + b\sigma_x + c\sigma_y + d\sigma_x\sigma_y$. Or equivalently, it is easy to see that, by translating and subtracting the first and the third template, an arbitrary template

p_{00}	p_{10}	p_{20}	p_{30}
	p_{11}	p_{21}	
p_{02}	p_{12}		

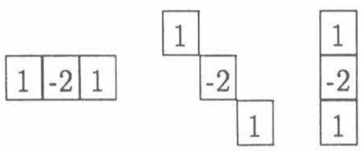
can be reduced to a 2×2 square

$$\begin{array}{|c|c|}\hline p_{00} & p_{10} \\ \hline p_{01} & p_{11} \\ \hline\end{array} \tag{5}$$

But, according to Theorem 4, the polynomials of K_{22} form a basis for I_{22}. From the determinantal expression

$$\begin{vmatrix} 1 & x_1 & y_1 & \sigma_x^{x_1}\sigma_y^{y_1} \\ 1 & x_2 & y_2 & \sigma_x^{x_2}\sigma_y^{y_2} \\ 1 & x_3 & y_3 & \sigma_x^{x_3}\sigma_y^{y_3} \\ 1 & x_4 & y_4 & \sigma_x^{x_4}\sigma_y^{y_4} \end{vmatrix}$$

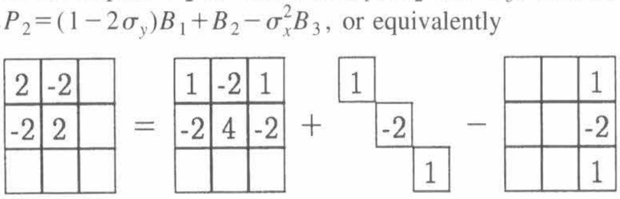
for the polynomials of K_{MN}, it follows that $K_{22}=\{P_2\}$. Hence, R must be a multiple of P_2, and therefore every polynomial P of K_{MN} has a decomposition of the required form, i.e. with all terms A_iP_i in I_{MN}. In other words, given a $M\times N$ rectangular region, any template in K_{MN} can be decomposed into the elementary templates [Eq. (4)] without leaving the rectangle. This is not always possible. Suppose we use the templates



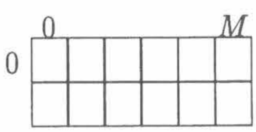
(6)

to detect the linearity of f. These are 1-D templates; they verify the linearity of f in the vertical, horizontal and one of the diagonal directions. The corresponding polynomials are $B_1=(\sigma_x-1)^2$, $B_2=(\sigma_x\sigma_y-1)^2$, and $B_3=(\sigma_y-1)^2$. One can easily see that the polynomials P_i and B_i generate the same ideal I, hence their templates are recognizing the same kind of feature. It follows that we can use these three new 1-D templates to detect the same 2-D feature, but not without introducing additional errors.

In fact, it is clear that the 2×2 square [Eq. (5)] cannot be reduced any further without traversing its boundaries. However, since both bases generate the same ideal I, we can decompose P_2 in terms of B_1, B_2 and B_3; that is, $2P_2=(1-2\sigma_y)B_1+B_2-\sigma_x^2B_3$, or equivalently



Hence, if we have a rectangular $M\times 2$ region



or a rectangular $2\times N$ region then there are templates in K_{MN} that cannot be decomposed into a combination of the templates of Eq. (6) without leaving the rectangle.

7 Concluding Remarks

In this paper, we proposed a method to convert differential operators into difference operators that can be applied to digitized functions. The main idea is to replace a differential operator by a decision tree with feature detectors and difference operators. We gave formal results that guide the design of such a tree. We discussed the choices that must be made, and we gave error bounds for the performance of the difference operators. Groebner bases turned out to be powerful tools to examine the mathematical structure of feature detectors and systems of difference equations. They can be used to reduce a system of difference equations into its simplest form, and to decompose the templates of the feature detectors into elementary templates.

We have given a number of practical examples to illustrate the design aspects and the practical use of decision trees. According to these examples, the combined use of feature detection and difference operators based on firm theoretical results is possible and seems to be a promising technique.

8 Appendix

8.1 *Proof of Lemma 1*

If $R\tilde{g}=Q\tilde{g}$ for all $\tilde{g}\in G$, then $(R-Q)\tilde{g}=0$ for all $\tilde{g}\in G$. Furthermore, G is a linear subspace of the vector space of all real functions $h:\mathbf{R}^m\rightarrow\mathbf{R}$. Every difference equation $P_i\tilde{g}=0$ states that $\tilde{g}$ must be orthogonal to some vectors $\mathbf{p}_i^k$, where the coordinates of the vector $\mathbf{p}_i^k$ are the coefficients of P_i shifted over k places, $k\in\mathbf{Z}$. Conversely, every vector that is orthogonal to G can be written as a linear combination of the vectors $\mathbf{p}_i^k$. It follows that any operator $(R-Q)$ can be written as $(R-Q)=\Sigma A_iP_i$.

8.2 *Proof of Theorem 4*

Since $\tilde{g}=\alpha_1\tilde{g}_1+\cdots+\alpha_l\tilde{g}_l$, the system of Eq. (1) can be rewritten as

$$\begin{cases} |[\alpha_1\tilde{g}_1(x)+\cdots\alpha_l\tilde{g}_l(x)]-f(x)|\leqslant\epsilon \\ \sum p_i[\alpha_1\tilde{g}_1(x+i)+\cdots+\alpha_l\tilde{g}_l(x+i)]=0. \end{cases} \quad (7)$$

Because the functions $\tilde{g}_j$ are solutions of the difference equations $P_i\tilde{g}=0$, the equalities of this system are trivially true. The legal linear dependences of this system that involve precisely $l+1$ of the inequalities correspond to the polynomials P of K_D. According to the elimination theory for linear inequalities, the validity of these dependences is sufficient for the system to be solvable.[18] Therefore, the system of Eq. (7) has a solution if and only if $|Pf|\leqslant\epsilon\Sigma|p_i|$ for all P of K_D. Furthermore, every legal linear dependence for Eq. (1) generates a similar legal linear dependence for Eq. (7), and vice versa. Hence, the polynomials of K_D form a basis for I_D.

References

1. B. O'Neill, *Elementary Differential Geometry*, Academic Press, New York (1966).
2. C. Dodson and T. Poston, *Tensor Geometry*, 2nd ed., Graduate Texts in Mathematics 130, Springer-Verlag, New York (1991).
3. A. Gray, *Modern Differential Geometry of Curves and Surfaces*, CRC Press, Boca Raton, FL (1993).
4. M. Brady, "Computational approaches to image understanding," *Comput. Surv.* **4**(1), 3–71 (1982).
5. V. Torre and T. A. Poggio, "On edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.* **8**(2), 147–163 (1986).
6. J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.* **8**(6), 679–698 (1986).
7. S. Sarkar and K. L. Boyer, "On optimal infinite impulse response

edge detection filters," *IEEE Trans. Pattern Anal. Mach. Intell.* **13**(11), 1154–1171 (1991).
8. R. Mehrotra and S. Zhan, "A computational approach to zero-crossing-based two-dimensional edge detection," *CVGIP Graph. Models Image Process.* **58**(1), 1–17 (1996).
9. R. Haralick, "Digital step edges from zero crossing of second directional derivatives," *IEEE Trans. Pattern Anal. Mach. Intell.* **6**(1), 58–68 (1984).
10. V. S. Nalwa and T. O. Binford, "On detecting edges," *IEEE Trans. Pattern Anal. Mach. Intell.* **8**(6), 699–714 (1986).
11. M. Fleck, "Some defects in finite-difference edge finders," *IEEE Trans. Pattern Anal. Mach. Intell.* **14**(3), 337–345 (1992).
12. M. Fleck, "Multiple widths yield reliable finite differences," *IEEE Trans. Pattern Anal. Mach. Intell.* **14**(4), 412–429 (1992).
13. P. Saint-Marc, J.-S. Chen, and G. Medioni, "Adaptive smoothing: a general tool for early vision," *IEEE Trans. Pattern Anal. Mach. Intell.* **13**(6), 514–529 (1991).
14. P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Mach. Intell.* **12**(7), 629–639 (1990).
15. D. Cox, J. Little, and D. O'Shea, *Ideals, Varieties and Algorithms: an Introduction to Computational Algebraic Geometry and Commutative Algebra*, Springer-Verlag, New York (1992).
16. J. Stoer and C. Witzgall, *Convexity and Optimization in Finite Dimensions I*, Springer, Berlin (1970).
17. P. Veelaert, "Digital planarity of rectangular surface segments," *IEEE Trans. Pattern Anal. Mach. Intell.* **16**(6), 647–652 (1994).
18. P. Veelaert, "On the flatness of digital hyperplanes," *J. Math. Imag. Vis.* **3**(2), 205–221 (1993).

Peter Veelaert received a degree in electrical engineering in 1981 and a PhD in electrical engineering in 1992, both from the University of Ghent. From 1981 until 1984 he was an engineer at the Digitized Information Systems Corporation, Belgium. He currently holds a postdoctoral position at the University of Ghent. His research activities include digital image processing and mobile robot navigation.