Measurement and characterisation in vision geometry

A. W. M. Smeulders* L. Dorst+ M. Worring*

(*) Intelligent Sensory Information Systems and
(+) Intelligent Autonomous Systems,
University of Amsterdam
Kruislaan 403, 1098 SJ AMSTERDAM, the Netherlands
{smeulders, leo, worring}@wins.uva.nl

Abstract

We discuss measurement of properties in digitised images. We give an overview of the most accurate as well as practical feature estimation methods, particularly of geometry measurement on straight lines and circular arcs. The theory offered here gives an upper bound to the accuracy of measurement and characterisation of any figure due to digitisation.

1 INTRODUCTION

In this paper we consider the task of measurement. Even here, the vision task can in most cases not be seen independently of its context. For intensity value measurement, the trustworthiness of the result is determined by the quality of the sensor and by the success in segmenting the target. For the measurement of texture, loosely defined as the interplay of geometry and intensity value patterns, the definition of features is not physically motivated, and hence no universal definition exists. In fact, texture is a context specific, purposeful property of an object's picture. Few references are known on the effect of the digital grid on texture measurement. This can be understood from the problem specific use of texture, for example [11]. Texture is not discussed here.

Measurement of geometry has a context free definition, apart from murky details such as the fractal character of a coast line. Length, curvature, area and related features have a unique and universal definition in the continuous world. In this paper, we discuss the fundamental bounds which arise when going from the idealised continuous world to the digital world.

In this communication we limit ourselves to context free measurement of image properties of noise free images. We aim at formulating bounds in the accuracy of measurement due to the digitization.

SPIE Vol. 3168 • 0277-786X/97/$10.00

2 TO MEASURE DIGITISED FIGURES

When measuring from a digitised figure, the ground truth is in the *continuous measurement* from the *continuous figure*, i.e. the figure before it is digitised. The task at hand is to design a digital measurement method working on the digitised figure in such a way that the outcome approximates the continuous measurement as faithfully as the digitisation permits.

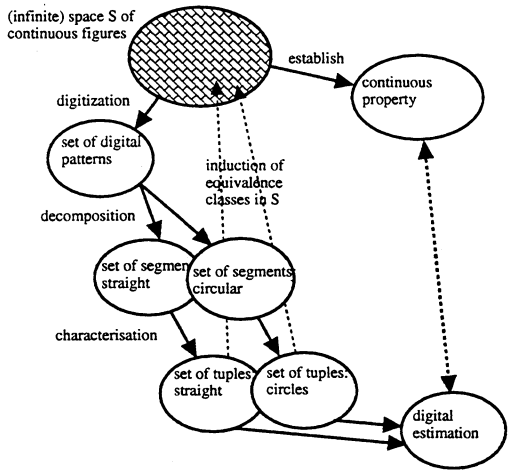

Figure 1: Measurement of digitised figures: the general scheme. Note that digitisation plus subsequent characterisation induce equivalence classes of object which cannot be discriminated by digital measurement.

In figure 1 the principle of digital measurement is laid down. A continuous figure S which may have infinitely many shapes and forms is measured in one of its properties. Usually, for geometrical properties such as length, curvature, and various shape criteria, the spectrum of outcomes is continuous; the measurement of the property can assume many values.

Due to the digitisation a pattern P of digital points arises, representative for the continuous figure. The *complement* of P is pattern P^*, the set of pixels excluded from the digitized figure. In the history of digital image processing it took some time to realise that any geometry is a collection of points rather than the lines or squares connecting them. The immediate consequence is that in the digital world there is no natural definition of spatial coherence, but that connectivity is a computational choice. If the choice is 8-connectivity for P, then P^* must be 4- connected to provide topological consistency. In the hexagonal grid the connectivity of figure and background is 6 both, and this is the reason why it has been advocated in the early days, mostly by Serra. However, the generalisation to the 3D-grid has silenced the call for the 'natural' hexagonal grid, as it is technically more cumbersome to generate sensors, displays and

basic computational algorithms. Square grids or at least rectangular grids are the only ones to survive today. We will conceive of connectivity as the subset of the P_i included by the digitisation, and the complement P_I^* excluded from the digitisation.

The digitisation induces a partitioning in the set of continuous figures. Some figures, very similar to one another, have the same digital pattern P_i and hence P_i^*. The set of continuous figures inducing the same pattern P_i is called the *locale* R_i [12]. A useful representation of R is found by a (generalised) Hough-transformation of the spatial domain into a parameter-domain. The transformation let a standard shape S be represented by a point. The edges of R_i in parameter space are the mapping of the grid points; an illustration will follow for the straight line and circular case.

Given the digitisation the contour of the figure is decomposed into segments of standard shape. In this paper we limit ourselves to decomposition into straight line segments or circular arcs. In the sequel we give examples of decomposition algorithms for that purpose.

Once the digital pattern is decomposed, we characterise the pattern P_i by a tuple of parameters. A characterisation C_i is a parameter representation of P_i such that one or more patterns P_i are mapped onto one value of C_i. As a consequence, the locale R_{C_i} encompasses the locales of one or more digitisation patterns R_{P_i}. The characterisation C_i is said to be complete when there is a one-to-one mapping between C_i and P_i. If the characterisation is incomplete, information about the precise shape and position of the continuous figure is lost (in addition to the information loss at the digitization). Most practical computer measurement algorithms have such an information loss as an effect, mostly because the characterisation is easy to compute. We will discuss complete characterisation schemes as well as several easy to compute incomplete characterisation schemes.

Usually, finding a characterisation is specific for the basic figure: a line or a circle. In the sequel we consider them, and the analysis is remarkably different. Veelaert [22] has presented a scheme for the characterisation of arbitrary bounding curves by sets of equations of inequality, one to each point of the grid.

From the point of measurement of the property, the most precise characterisation is the complete characterisation. This bound due to digitisation induces a bound in the digital measurement accuracy. It is the deterministic equivalent of the Cramér/Rao bound known from stochastic parameter estimation theory. For each pattern P_i, only one estimate $\hat{g}$ of the property at hand can be assigned. As the true value of the property g varies over the corresponding locale R_i, an uncertainty is unavoidably introduced. This is quantified in the domain variance [28]:

$$V_{\hat{g}}(P_i) = \int_{R_i} p(S/P_i)(\hat{g}(P_i) - g(S))^2 dS, \tag{1}$$

where $p(S/P_i)$ is the probability density that the continuous figure S has lead to pattern P_i. We seek the estimator $\hat{g}_{mv}(P_i)$ minimising the domain variance:

$$\hat{g}_{mv}(P_i) = \arg \min_{\hat{g}} V_{\hat{g}}(P_i) \tag{2}$$

The solution is equal to the expectation of g over R_i.

$$\hat{g}_{mv}(P_i) = \int_{R_i} p(S/P_i)g(S)dS \tag{3}$$

For a proof see the reference. As the domain variance is a deterministic measure, the domain variance of $\hat{g}_{mv}$ is the minimal variance one can reach for any possible estimator: the *geometric minimum variance bound*, GMVB:

$$G_g(P_i) = V_{\hat{g}_{mv}}(P_i) \tag{4}$$

The bound quantifies the ability of the pattern P_i to discriminate among variations in the property of the continuous figures. It expresses the maximum achievable precision in property estimation after digitisation.

To actually compute an unbiased, minimum variance estimate of a property consider the moment generating function for g depending on parameters describing the set of all continuous figures $\mathbf{s}$:

$$T_g^i(P_i) = \int_{\mathbf{S}} \tilde{p}(\mathbf{s}/P_i)g^i(\mathbf{s})d\mathbf{s}, \tag{5}$$

where the function $\tilde{p}(\mathbf{s}/P_i)$ is proportional to the conditional probability function of figure $\mathbf{s}$ given P_i. Now:

$$\hat{g}_{mv}(P_i) = \frac{T_g^1(P_i)}{T_g^0(P_i)} \tag{6}$$

and

$$G_g(P_i) = \frac{T_g^2(P_i)}{T_g^0(P_i)} - \frac{T_g^1(P_i)}{T_g^0(P_i)} \tag{7}$$

3 STRAIGHT LINES AND THE GRID

3.1 Digitisation

The digitisation of a straight line $S_i : y = \alpha x + e$ is a pattern P_i of pixels with:

$$(x_i, \lfloor \alpha x_i + e \rfloor) \tag{8}$$

In this x_i is at integer, grid column, positions. This definition of digitisation is known as the 8-connected object boundary quantization. The 4-connected or the 6-connected grid will create an identical structure, see [25]. The digitisation scheme renders the pixels on and just below the line. The pixels just above the line are in P_i^*. In the sequel we only consider lines in the first octant of the grid. The rest follows by symmetry.

The digitisation pattern may be coded by assigning 0 for each displacement in the positive x-direction of the grid, and 1 for each displacement in the diagonal direction. A string is said to be straight iff there exists a straight continuous

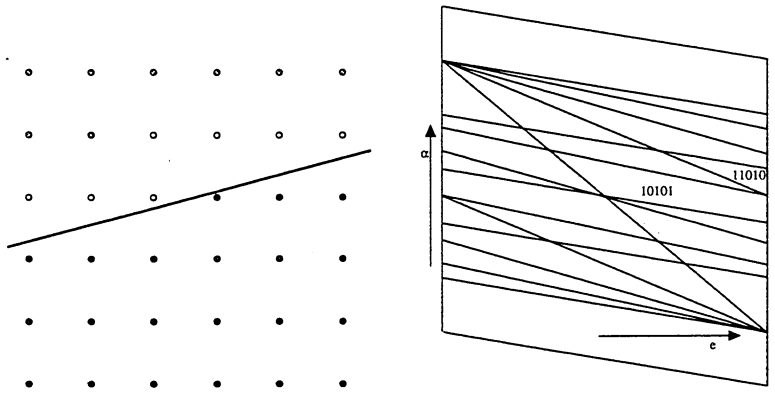

Figure 2: (a) Regular grid of pixels in the spatial domain for string $P_i = 00100$, represented by filled pixels. Open pixels are in the complement P_i^*. (b) Same figure, shown in (e, α)-domain, first octant.

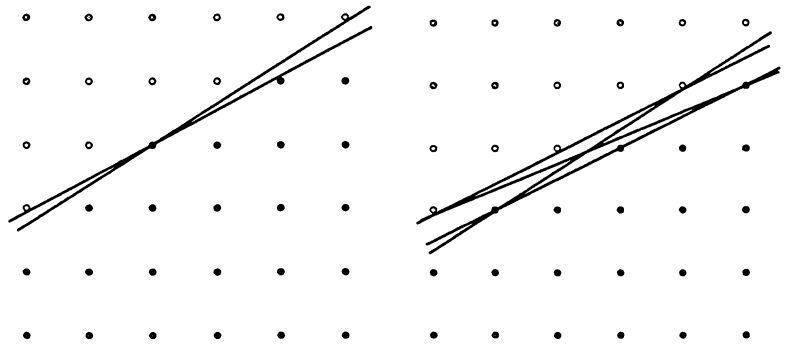

Figure 3: The code string 11010 and 10101 on the grid. The locale R_{11010} is a triangle in the (e, α)-domain and a diamond for R_{10101}, see figure 2.

line from whom it could have been generated by digitisation. An example of a straight string is 10100101, whereas an example of a non-straight string is 11110000.

The locale R_i of P_i is bounded by lines with maximum and minimum slope just bouncing against the extreme grid points in P_i and P_i^*. In general, there are 4 such lines. We consider locales in the (e, α)-domain which is the generalised Hough transform of a line to its parameters [25]:

$$y_i = \alpha x_i + e \Longleftrightarrow e = -x_i \alpha + y_i \tag{9}$$

In that domain, the locales have no more than two different shapes. In principle we expect them to be diamond shaped bounded by 4 edges, one to each of the 4 limiting grid points. Invariably, the points to the East and to the West share the same value of α. For this value, α_w, the diamond is at its widest. It indicates the value of α where the continuous line can be shifted up and down with maximum degree of freedom in e while still generating the same code, see figure 3. The degeneration of a diamond is a triangle. They are oriented in such a way that one point of the triangle holds the middle value of α, with $\alpha = \alpha_w$.

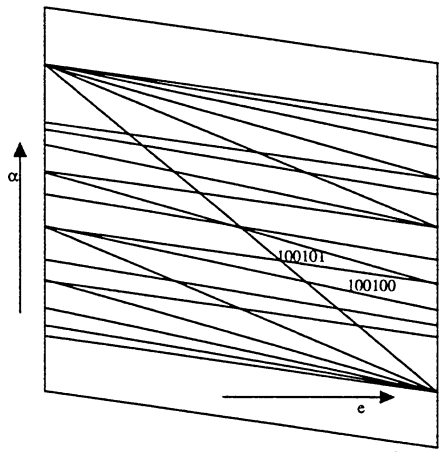

Figure 4: Regular grid of 7x7 pixels in (e, α)-domain, first octant shown.

If we compare in the (e, α)-domain a grid of 6 x 6 pixels to an extended grid of 7 x 7 pixels, we make two observations. First, it can be seen that the newly added lines in (e, α)-domain cut existing locales into one diamond and one triangle. The R_{10010} is cut into a diamond for 100100 and a triangle for 100101. This can be explained later on from the fact that a straight extension to the existing string one has an established periodicity and the other one just became a-periodic. Secondly, not all locales are affected by the extension. The shape of R_{010110} is identical to the shape of R_{01011}. This can only happen when the alternative extension, 010111, is absent due to the fact that the string no longer constitutes a straight string.

A string P, or a digitisation pattern for that matter, is said to be periodic when a substring P_q, the *period*, in P can be identified such that P contains repetitions of P_q and no other elements. A string P is said to be quasi-periodic when substring P_q can be identified such that a tail-end of P_q added to the head of P, and a head-end of P_q added to the tail of P will turn P into a periodic string. When there is a choice, we select as P_q the shortest string. The string $P = 10100101010$ is a quasi-periodic string as a substring $P_q = 0010101$ can be identified of which head-end 0010 and tail-end 010101 turn P into the periodic string 0010— 10100101010— 010101. Even a-periodic strings such as 10001001 can be quasi-periodic as in this case P_q is 0001001.

3.2 Decomposition

To verify whether a string is a straight string was first considered by Brons [6] in narrative terms. These linearity conditions were formulated more precisely by Wu as follows [29]:

Start at $P_i^{d=0} = P_i$:
⇓
Number of different symbols in $P_i^d > 2$? Stop, not a straight string.
Number of different symbols in $P_i^d = 1$? Stop, straight string.
⇓
Do the symbols in P_i^d differ by one? No: Stop, not a straight string.
⇓
Does one of them occur singly? No: Stop, not a straight string.
⇓
Does the symbol occurring singly appear once? Yes: Stop, straight string.
⇓
Compute the string P_i^{d+1} of run lengths between two minority codes of P_i^d.
Is the first element of $P_i^{d+1} > \max(P_i^{d+1})$, or,
is the last element of $P_i^{d+1} > \max(P_i^{d+1})$? No: Stop, not a straight string.
⇓
Is the first element of $P_i^{d+1} < \max(P_i^{d+1})$? Yes: Remove the first element from P_i^{d+1}.
Is the last element of $P_i^{d+1} < \max(P_i^{d+1})$? Yes: Remove the last element from P_i^{d+1}.
⇓
Loop back to the beginning and verify P_i^{d+1}.

By looping through the conditions one establishes the condition of straightness. The multi-level testing is needed as not any string with 0's and 1's is straight. Obviously 000001111 is not a straight string. Also 000100101 is not a straight string as the string on the next level of consideration is 432 containing more than two elements which in addition differ more than 1 in value. Finally, 010101001010100101001 is not a straight string. Where it appears to be straight on the second level 222322323, it is not on the second level, 432.

A more interesting question is the straightness of a *growing* string. For example, when an arbitrary digitisation pattern is followed, how to decompose the string in segments of maximum length while each segment is straight still?

To that end, several algorithms have been designed.

Lindenbaum [15] considers a string ${}_nP$ of n-elements. Then he checks what the locale ${}_nR$ of ${}_nP$ is and compares this with ${}_{n-1}R$ of string ${}_{n-1}P$. Iff ${}_nR \subseteq_{n-1} R$ then ${}_nP$ is a straight extension of ${}_{n-1}P$.

In an algorithm described in [20] the tests of Wu are verified sequentially while book keeping the essential string parameters at each level. At each new level the incoming element is verified against the minority and majority value at that level, until one reaches the highest level where the minority value has not been defined yet. So the algorithm in the reference stores all relevant information on the growing string. For each level d: the value of the majority element M_d, the value of the minority element m_d, the value of the initial element i_d only kept for completeness, and, the value of the current, running element r_d. As an example, for string $P_i^0 = 0101010010101001010$ the $M_0 = 0$, $m_0 = 1$ and $r_0 = 0$. At level $d = 1$, $M_1 = 2$, $m_1 = 3$ and $r_1 = 1$, a summary of P_i^1 =22232232. At level $d = 2$, $P_i^2 = 431$ with $M_2 = 4$, $m_2 = 3$ and $r_2 = 1$, or $M_2 = 3$, $m_2 = 4$ and $r_2 = 1$ as it has not been decided yet which is the minority element and which one is the majority one. The running element of level d forms an element at one level higher when r_d equals m_d. Checking will proceed to one level higher only when r_d reaches this value. That is why, for most elements, the checking for straightness will not pass beyond level 1 or 2. Only where minority elements at all levels coincide the depth of iteration over the levels is fully needed. The algorithm is proved in the reference with estimates for its complexity.

A geometrical interpretation of this string testing scheme follows from running down a straight string. Running down a string implies that the size of the grid is expanded by one column, see figure 2. In the (e, α)-domain this is a transition into figure 4. For string ${}_nP$, found to be straight so far, the effect on its locale ${}_{n+1}R$ is that the line corresponding to the new grid column cuts ${}_nR$ into two, or that ${}_nR$ is left unchanged: ${}_{n+1}R =_n R$. If ${}_{n+1}R$ is unchanged there is no need to check the boundaries of the R, hence no recursion on the levels is needed.

3.3 Characterisation

A straight string can be completely characterised by 4 integer parameters (N, p, q, s), where N is the number of codes in the string equal to the number of points on the grid minus 1. The s is a phase factor indicating the position in the string where the first P_q starts. In fact, s plays a marginal role in any computation. The parameters q and p are the length of period P_q and the number of minority elements therein, respectively. The period P_q has been brought in standard form such that it always starts with all majority elements first and is concluded by the minority element. The value of the most prevailing α where the locale is at its widest has value $\alpha_w = \frac{p}{q}$. For a proof, see [7].

The boldest characterisation is to reduce the information in the pattern of P and count the number of codes, $C = (N)$, equal to the number of digitisation points in the string minus 1. That boils down to taking all locales together and integrating over all of the area in the (e, α)- domain.

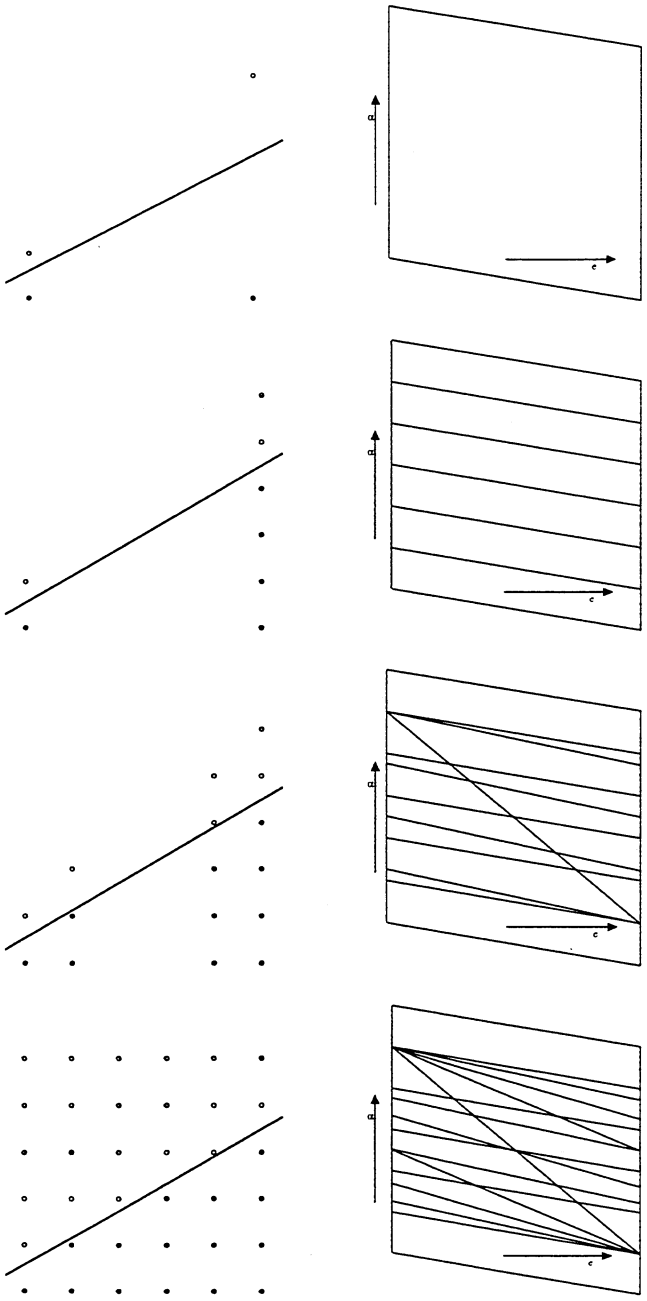

Figure 5: (a) Points of interest in the characterisation by (N) in the spatial domain and the corresponding locale in (e, α)-domain. (b) Characterisation by (N_0, N_1). (c) Characterisation by (N_0, N_1, N_c). (d) Characterisation by (N, p, q, s).

A straight string can also be characterised by the parameters $C = (N_0, N_1)$ being the number of codes in horizontal and vertical direction, and the number of codes in diagonal direction. This characterisation, in fact, determines at which position the string exits the grid. Each point in the exit column of grid points now generates a different value of the characterisation.

A final characterisation of the string is by adding as a third parameter the number of changes between a diagonal and a non-diagonal code. The characterising tuple in this case is $C = (N_0, N_1, N_c)$. For example, $C_{N_0,N_1,N_c} = (3, 2, 3)$.

3.4 Measurement of properties: example length

When considering measurement of say the length, ℓ, of a line, even today many text books write:

$$\ell = N_0 + \sqrt{2}N_1, \tag{10}$$

If anything, this is the length of the broken line connecting the digital points, definitely not a proper estimate for the length of the continuous line before it was digitised.

For proper measurement, the function we need to include is the probability of the lines. Rather than considering a uniform distribution in (e, α), it is better to do so in the parameters of the normal representation as that corresponds to tossing lines randomly on a co-ordinate system. The density function of the probability of a line with (e, α) can be found by a co- ordinate transformation. It yields [8]:

$$p(e, \alpha) = c(1 + \alpha^2)^{-\frac{3}{2}} \tag{11}$$

with c follows from normalisation. This would yield $c = \sqrt{2}$, but as the locales of the digitised lines are diamond shaped rather than square it holds that,

$$c = \frac{1}{\sqrt{n^2+1} - n + \frac{1}{2}\sqrt{2}\sqrt{2n^2+2n+1} - (n\sqrt{2} + \frac{1}{2}\sqrt{2})}. \tag{12}$$

After extensive calculations, Beckers [3] gives the probability density function for lines in 3D.

Once p(S) is known, equation 4 can be evaluated for each different characterisation. The most accurate result is found for the complete characterisation, in the case of straight lines given by the tuple (N, p, q, s). They have been derived in [8] in mathematical expressions and tabulated form for length and orientation. No analytic solution is known, however, and even the Taylor expansions take more length than is available here.

So, as it is unpractical to use the locales for each digitisation pattern P, we turn to the use of characterisation C_i of P_i. Computing the unbiased length estimate for the (N)-characterisation, one finds:

$$\ell = 1.111N, \tag{13}$$

The residual error when considering a line of random orientation with respect to the grid is computed as 11.2%. That percentage is the difference between

the length of the original line and the outcome averaged over all orientations. The formula implies that counting the number of edge points and taking that for the length gives a bias of 11.1%.

The characterisation by (N_0, N_1), gives as its best length:

$$\ell = 0.945N_0 + 1.346N_1, \tag{14}$$

an equation closely resembling Kulpa [13]. The residual error when considering a line of random orientation with respect to the grid is computer as 2.6%. The formula was found in slightly different form by Profitt and Rosen [18].

Note that computing the distance between the discrete begin and end point of a straight line also employs the same characterisation. Only the length estimator is different:

$$\ell = \sqrt{(N_0 + N_1)^2 + N_1^2} \tag{15}$$

This formula is due to Pythagoras [19]. It is cited here to stipulate that (1) the above counting formulae are linear in the amount of steps rather than quadratic, and (2) in the digital world measuring length is *not* the same as establishing distance between end points. For the latter, one cannot do better than the Pythagorian formula, whereas for the measurement of length one can do better by taking into account the configuration of points in between.

The characterisation by (N_0, N_1, N_c) is much finer than the previous two. The resulting unbiased estimate is given by:

$$\ell = 0.980N_0 + 1.406N_1 - 0.091N_c \tag{16}$$

with residual error of 0.8% [25].

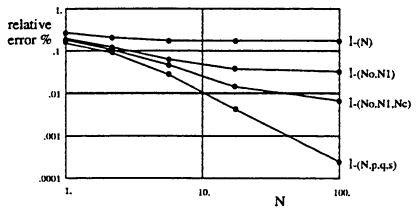

Figure 6: The accuracy of length estimators in 2D. From [9].

In the figure 6 the accuracy of each of the length estimators is displayed for varying length of the string. A few observations can be made. (1) The accuracy of the linear estimates based on the (N), (N_0, N_1) or (N_0, N_1, N_c) characterisations levels off for increasing values of N. No matter how dense the figure is sampled, the same accuracy will result. This is due to the bias depending on the direction of the figure to the grid. (2) The accuracy of the squared estimator increases quadratically as expected for the continuous case. (3) The accuracy of the most precise characterisation falls off with a greater

accuracy than the Pythagorian equation. The digital grid helps confining the set of continuous figures.

The topic of measurement of properties from a linear characterisation has been extensively studied, each author contributing his own case. Mulliken gives an unbiased estimate for computing the surface in 3D [17]. Beckers contributes unbiased estimates for the length in 3D [4]. Verwer [24] gives a numerical evaluation for length estimates in any dimension.

4 CIRCLES AND THE GRID

4.1 Digitisation

We now turn to circular figures [28]. Again, let S represent a continuous figure. Given the digitised pattern P_i derived of S. The approach is to identify the set S of all continuous circular arcs which share P_i as their digitisation pattern, see figure 7. The continuous figures S are represented by a point each in the parameter space (x_M, y_M, r), yielding the centre positions and radii. The transform between the spatial representation of the figure and the parameter representation is the three dimensional generalised Hough transform:

$$(x - x_M)^2 + (y - y_M)^2 - r^2 = 0 \Longleftrightarrow (x_M - x)^2 + (y_M - y)^2 - r^2 = 0 \quad (17)$$

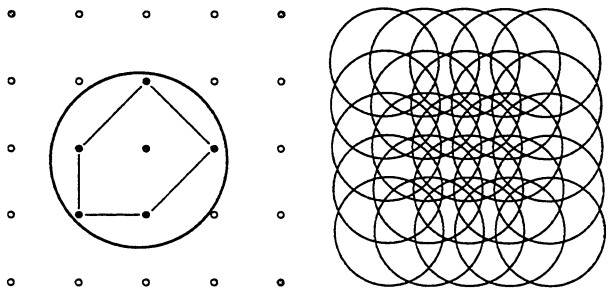

Figure 7: (a) A digitisation pattern P_i as it originates of some (unknown) continuous circular arc S. (b) The parameter space for circles shown for fixed r. Indicated are the boundaries of the locales, images of the points of the grid.

Varying the parameters of (x_M, y_M, r) one finds bounds beyond which the parameter cannot be changed without alterations to the digitisation pattern P_i. Figure 8 gives an example of the bounding circles of a fixed radius of P_i and of P_i^*.

In P_i or P_i^* not all pixels matter in finding the extent of R_i. Some of the pixels in these two sets will not be part of the set of bounding pixels, not for any value of r. Some of the pixels in these sets will be part of the bounding

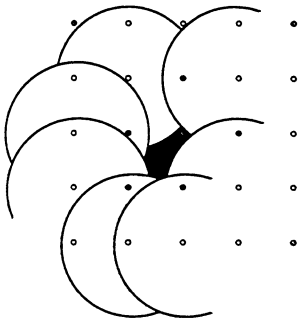

Figure 8: The locale for fixed radius in (x_M, y_M)-space.

pixels. Occasionally, some pixels of P_i or P_i^* will be part of the bounding set of pixels for all r.

In (x_M, y_M, r)-space, for fixed r' a pixel P_1 in the grid transforms to a circle with radius r' centred around $(x_M = x_{P_1}, y_M = y_{P_1})$. That circles constitutes the set of all centre points in the spatial domain of which the circle will pass through P_1. Hence, as pixels of the grid bound the circles fitting between the pixel set P_i on the inside and P_i^* on the outside must be bounded in parameter space by circular arcs. For fixed r' , the locale $R_i(r')$ cut out by digital pattern P_i is one (or more) polygon(s), curved in its edges with radius r'. Each pixel in P_i corresponds to one convex edge in $R(r')$, whereas each pixel in P_i^* corresponds to one concave edge in $R_i(r')$, and there are no other edges in $R_i(r')$. Vertices in $R_i(r')$ are determined by two points of $\{P_i, P_i^*\}$. The combination of all $R_i(r)$ while varying r gives a three dimensional, connected structure R_i. The example in figure 9 gives rise to the following curved edged polygons:

r	which points determine $R_i(r)$
00.0	$\emptyset$
16.8	p_0, p_3, p_2
17.0	p_0, p_3, p_2, p_1
18.5	$p_0, p_3, p_1^*, p_2, p_1$
19.7	$p_0, p_3, p_1^*, p_2, p_2^*, p_1$
20.2	$p_0, p_1^*, p_2, p_2^*, p_1$
22.5	p_0, p_1^*, p_2^*, p_1
25.5	p_0, p_1^*, p_2^*
31.6	$\emptyset$

In the reference, a precise account is given for all transitions in the edges of $R_i(r)$ while increasing r following the properties of the α- hull [27]. It ap-

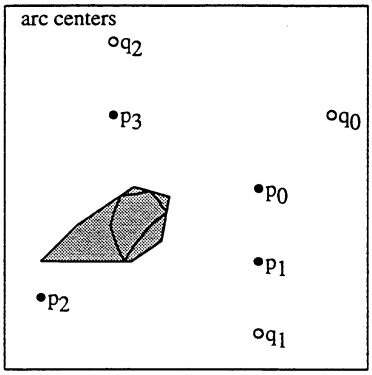

Figure 9: The projection of $R_i(r)$ onto the (x_M, y_M)-plane is a Voronoi diagram. (Elements of P_i^* indicated by q_i.)

pears that there is a limited number of cases which occur, corresponding to the configuration of bounding pixels points. The projection of $R_i(r)$ onto the (x_M, y_M)-plane is the generalised Voronoi polygon.

4.2 Decomposition in curves

Sofar, no use has been made of the properties peculiar to the regular grid. We just considered point sets P_i included in the digitisation, and P_i^* excluded by the digitisation. On a regular grid, there are six different shapes for the projection of R_i on the (x_M, y_M) plane. They can be classified as straight, strictly convex, infinitely convex, strictly concave, infinitely concave or non-circular. Each of the cases is illustrated by an example in figure 10. The decomposition of an

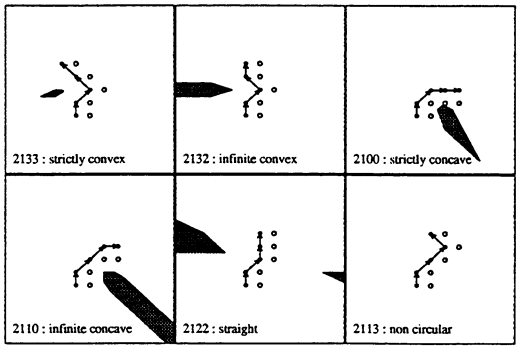

Figure 10: Examples for each of the 6 different shapes of the projection of R_i on the (x_M, y_M) plane on a regular grid. From [28].

arbitray string $\partial_N P_i$ of length N can be done by checking whether the locales ${}_N R_i$ has a non-zero intersection with ${}_{N-1} R_i$.

4.3 Characterisation and measurement of curves

In literature no methods of characterisation are known, apart from the complete characterisation of a circle given above. Usually characterisation and measurement is done is one pass. For the estimation of the position and radius of curves of the digitised boundary, five methods were compared in [26]. It turned out that none of the methods yields accurate and precise estimates of curvature. A theoretical analysis of the methods revealed a number of clues for the poor performance. In [16] the curvature is estimated by applying a linear differentiating filter to the x- and y-co-ordinate separately. For circular arcs this separation, overlooking the 2D-nature of the problem, introduces significant errors due to the truncation of the filter. Some methods [1, 2, 10] find curvature by applying a linear differentiating filter to the estimated orientation. The method in [10] is the only method which explicitly takes the anisotropy of the grid into account and in fact with proper scaling yields the best performance. In the other cases errors in the order of 40% are common. Fitting a circular arc to the digital data does take the two dimensional shape of the digital boundary into account and algorithms can be found in [5, 14, 21]. It turned out that for noise free digitised circular arcs the precision is poor.

The best possible estimate is given by application of equation 4 in combination with the complete characterisation derived above. Results are given in [28], but only for small sampled sets in tabularized form. The results are illustrated in figure 11 by sketching the probability density function for this R_i as well as the result.

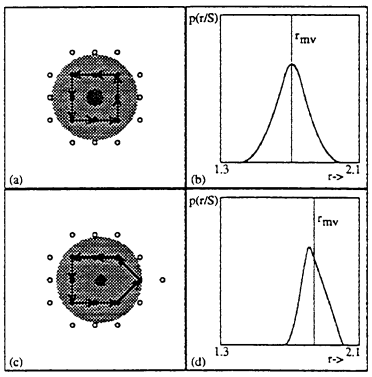

Figure 11: The edge of two patterns ∂P_i and their probability density function of the circles, with best estimate for the radius. From [28].

Based on the 6 different classes of patterns one can compute the maximally

recognizable radius $r_{min}(N)$ as a function of N, the number of points in ∂P_i. This value is the bound on the radius of continuous arcs one can estimate for each N. In effect, this provides a sampling theorem for curvature measurement. Practical bounds on the precision in measuring curvature indicate the relative deviation in the digitization limited optimal measurement of arcs with $r \leq 6$ grid units, using a window of $N = 10$ elements in ∂P_i is between 2% and 9%. For chains of 9 elements, a table for ∂P_i configurations can be formed of approximately 4500 entries. All other 33 million configurations of 10 pixels do not represent a circular arc. For digitizations of full disks, with varying N, the deviation is below 1% for $r \geq 4$ grid units.

As concerns the measurement of length, remarkably enough, the accuracy of the linear estimators in equations 13, 14, and 16, straight lines are worst compared to other possible curves. For any arc other than the straight line, the error in estimating its length is less than the straight case using the above formula (which have been optimised for straight lines!). This is due to the fact that an arc contains several values of α and hence compensates for the bias. As a consequence, for circular arcs approaching 22^o, the error in the length estimate even drops to 0 [9]. It makes the (N_0, N_1, N_c)-estimator a good candidate for measuring length of arbitrarily shaped curves.

5 TO MEASURE IDEAL BLACK/WHITE

Whereas length and other parameters of figures cannot be measured without loss of information from the digitized images we discussed so far, Van Vliet and Verbeek [23] have established error free measures in idealized recordings of continuous of black and white images. They start from Nyquist sampling and observe that the ideal black and white image will be transposed by the scanner into a grey valued image. Then, from this band limited image, taking the profile of the figure edge into account and knowing the transfer characteristic of the scanner, they derive estimates for iso-intensity curves of length, surface and curvature. These measures derived from ideally scanned grey value images are more accurate than the ones above with errors descending down to 0%.

6 DISCUSSION

The paper emphasizes the following points.

First of all, it took a long time to realise that establishing length in the digital world is a estimation problem with a limited accuracy more than anything else. Even where geometry is uniquely defined continuously or digitally, the outcome in a digital world can only be known with limited accuracy.

Secondly, measuring geometry from a digitized figure should not be done simply by filling in discrete parameters in the formulae of the continuous domain. The selection of the proper parameters to characterise such standard shapes as lines and circles resulting from digitisation is a non-trivial part of the process

to find estimators with optimal accuracy. An analysis is required in which way the proper parameters depend on the characterising parameters of the grid.

Thirdly, another way of saying is, that for noise free measurement the Cramér/Rao bound for a bound on the accuracy of continuous parameter estimation is not appropriate here. Instead, we derive the Geometric Minimum Variance Bound. We have shown the limited sets of pixels of the grid which are decisive in the ultimate accuracy of geometry measurement. In the case of straight these are 4 (or by coincidence 3) pixels, whereas in the case of circles this number may go up to infinity, be it that for the minimum radius smaller circles usually 4 or 5 pixels are sufficient.

Fourth, when measurement is not restricted to what is found in the (binary) image but the quality of the sensor is included in the analysis, the work by Van Vliet and Verbeek on grey-valued images gives a way to measure geometry with arbitrary accuracy.

References

[1] I.M. Anderson and J.C. Bezdek. Curvature and deflection of discrete arcs: A theory based on the commutator of scatter matrix pairs and its application to vertex detection in planar shape data. *IEEE trans PAMI*, 6:27–40, 1984.

[2] H. Asada and M. Brady. The curvature primal sketch. *IEEE trans. PAMI*, 8(1):2–14, 1986.

[3] A. L. D. Beckers and A. W. M. Smeulders. The probability of a line in 2 and 3 dimensions. *Patt. Recogn. Let.*, pages 233–240, 1990.

[4] A. L. D. Beckers and A. W. M. Smeulders. Optimal length and distance estimation in 3 dimensions. *Comp. Graph. Vis. Im. Proc*, 55(3):296–306, 1992.

[5] M. Brady and H. Asada. Smoothed local symmetries and their implementation. *The International Journal of Robotics Research*, 3(3):36–60, 1984.

[6] R. Brons. Linguistic methods for the description of a straight line on a grid. *Comp. Graph. Im. Proc.*, 2:48–62, 1974.

[7] L. Dorst and A. W. M. Smeulders. Discrete representation of straight lines. *IEEE–trans. PAMI*, 6:450–463, 1984.

[8] L. Dorst and A. W. M. Smeulders. Best linear unbiased estimates for properties of straight lines. *IEEE trans. PAMI*, 8:276–282, 1986.

[9] L. Dorst and A. W. M. Smeulders. Length estimators for digitized contours. *Comp. Graph. Vis. Im. Proc.*, 40:311–333, 1987.

[10] J. S. Duncan, F. Lee, A. W. M. Smeulders, and B.L. Zaret. A bending energy model for measurement of cardiac shape deformity. *IEEE Transactions on medical imaging*, 10(3):307–320, 1991.

[11] R. M. Haralick, K. Shanmugan, and I. Dinstein. Textural features for image classification. *IEEE trans. SMC*, 3:610–621, 1973.

[12] D. I. Havelock. Geometric precision in noise-free digital images. *IEEE trans. PAMI*, 11(10):1065–1075, 1989.

[13] Z. Kulpa. Area and perimeter measurement of blobs in discrete inary images. *Comp. Vis. Graph. Im. Proc.*, 6:434–454, 1977.

[14] U.M. Landau. Estimation of a circular arc center and its radius. *Comp. Vis. Graph. Im. Proc.*, 38:317–326, 1987.

[15] M. Lindenbaum and A. Bruckstein. On recursive, o(n) partitioning of a digitized curve into digital straight segments. *IEEE trans PAMI*, 15:949–953, 1993.

[16] F. Mokhtarian and A. Mackworth. Scale-based description and recognition of planar curves and two-dimensional shapes. *IEEE trans PAMI*, 8(1):34–43, 1986.

[17] J. C. Mulliken. Surface area estimation of digitized planes. *BioImaging*, 1:6–16, 1993.

[18] D. Profitt and D. Rosen. Metrication errors and coding efficiency of chain-encoding schemes for the representation of lines and edges. *Comp. Graph. Im. Proc.*, 10:318–332, 1979.

[19] Pythagoras. A theorem. *Trans. Greek Olympic Soc*, 15:1–20, 400 BC.

[20] A. W. M. Smeulders and L. Dorst. Straightness and characterisation of tracked arcs: a linear time algorithm. In P. Bhattacharya, R. A. Melter, and A. Rosenfeld, editors, *Vision Geometry*, pages 169–195. American Mathematical Society, Providence RI, 1991.

[21] S.M. Thomas and Y.T. Chan. A simple approach for the estimation of circular arc center and its radius. *Comp. Vis. Graph. Im. Proc.*, 45, 1989.

[22] P. Veelaert. Digital planarity of rectangular surface segments. *IEEE trans. PAMI*, 16:647–652, 1994.

[23] P.W. Verbeek and L.J. van Vliet. On the location error of curved edges in low-pass filtered 2-d and 3-d images. *IEEE trans PAMI*, 16:726–733, 1994.

[24] B.J.H. Verwer. Local distances for distance transformations in two and three dimensions. *Pattern Recognition Letters*, 12:671–682, 1991.

[25] A. M. Vossepoel and A. W. M. Smeulders. Vector code probability and metrication error in the representation of straight lines of finite length. *Comp. Graph. Im. Proc.*, 20:347–368, 1982.

[26] M. Worring and A. W. M. Smeulders. Digital curvature estimation. *Comp. Vis. Graph. Im. Proc: Image Understanding*, 58(3):366–382, 1993.

[27] M. Worring and A. W. M. Smeulders. Shape of an arbitrary finite point set in R^2. *Journal of Mathematical Imaging and Vision*, 4(2):151–170, 1994.

[28] M. Worring and A. W. M. Smeulders. Digitized circular arcs: characterization and parameter estimation. *IEEE Trans. PAMI*, 17(6):587–598, 1995.

[29] L. D. Wu. On the chain code of a line. *IEEE trans. PAMI*, 4:347–353, 1982.