Nonlinear Multiresolution: A Shape-from-Shading Example

Shmuel Peleg and Gad Ron

Abstract—**Multiresolution methods are often used in computer vision to speed up computationally intensive tasks. An approximate solution computed for a reduced image is used to guide the algorithm towards the complete solution on a larger image.**

Whenever the image is a nonlinear function of the parameters to be computed, multiresolution should be used with special care. The obvious image reduction is incorrect, and the solution on the reduced image is a bad estimate of the higher resolution solution.

Shape-from-shading is treated in this correspondence as an example of such difficulties. Reduction of the gray level resolution of an image, as often done in image "pyramids," does not correspond to images obtained by reducing the *shape resolution:* the 3-D resolution of an object. The correspondence between gray level and shape resolutions is therefore discussed, and a method is proposed for using multiresolution approach in the case of shape-from-shading.

It is concluded that a study of the relation between image resolution and estimated-parameter resolution should be done before using any multiresolution approach.

Index Terms—**Multiresolution, pyramids, shape-from-shading.**

I. Introduction

When image intensity is not a linear function of the parameters, as in the case of shape-from-shading, the obvious reduction of image resolution will not give optimal results. This correspondence studies one specific case of shape-from-shading, and analyzes the relation between shape resolution and image resolution.

Shape from shading methods try to reconstruct a 3-D surface from an intensity image when the surface reflectance properties are known. This correspondence treats surfaces with Lambertian reflectance properties.

In this section we will briefly review the Lambertian reflectance model and a proposed solution to the shape from shading problem, and describe the multiresolution pyramid.

A. The Lambertian Reflectance Model

Given a surface S and a point light source at direction $\vec{L}$, the surface is called *Lambertian* if the amount of light R reflected from the surface S is a function of the angle between the light direction $\vec{L}$ and the surface normal $\vec{N}_s$. It will be assumed that the reflectance coefficient of the surface is 1, and it will be omitted from all equations. The reflectance function of the surface S at every point can thus be written as follows:

$$R(\vec{N}_s, \vec{L}) = \cos(\alpha_{\vec{N}_s, \vec{L}}) = \frac{<\vec{N}_s \cdot \vec{L}>}{||\vec{N}_s|| \cdot ||\vec{L}||}. \quad (1)$$

By denoting $p = \frac{\partial S}{\partial x}$ and $q = \frac{\partial S}{\partial y}$, the surface normal can be written as $\vec{N}_s = (-1, p, q)$. Given a light source in the direction $\vec{L} = (-1, p_L, q_L)$, the Lambertian reflectance function at a point with surface derivatives (p, q) can be written as

$$R_{\vec{L}}(\vec{N}_s) = R_{\vec{L}}(p, q) = \frac{1 + p \cdot p_L + q \cdot q_L}{\sqrt{1 + p^2 + q^2} \cdot \sqrt{1 + p_L^2 + q_L^2}}.$$

Manuscript received September 10, 1989; revised June 14, 1990. Recommended for acceptance by R. Woodham. This work was supported by the Israeli NCRD. Part of this work was performed while S. Peleg was with David Sarnoff Research Center, Princeton, NJ.

The authors are with the Department of Computer Science, Hebrew University of Jerusalem, 91904 Jerusalem, Israel.

IEEE Log Number 9040024.

For a light source in the direction $\vec{L} = (-1, 0, 0)$ (illumination from above) the reflectance function is

$$R(\vec{N}_s) = R(p, q) = \frac{1}{\sqrt{1 + p^2 + q^2}}. \quad (2)$$

B. A Shape-from-Shading Algorithm

In shape-from-shading the inverse problem of the reflectance model is solved. Given a gray level image I of an object with a Lambertian surface, the surface S is to be reconstructed. This problem is underconstrained, and at any point the observed reflectance R can correspond to many (p, q) pairs. Brooks and Horn [1] added a smoothness constraint on the surface derivatives. They developed an iterative algorithm to compute the surface derivatives (p, q), minimizing the error between the observed intensity and the computed reflectance, giving weight to a smoothness term. The error term at every image point is

$$E(p, q) = (I - R(p, q))^2 + \lambda \cdot V(p, q),$$

where I is the given intensity at the point, R is the intensity computed from the suggested (p, q) using the reflectance model, $V(p, q)$ is the smoothness term for (p, q) at the point, and λ is the weight given to the smoothness term. The smoothness term was computed as

$$V(p, q) = \left(\frac{\partial p}{\partial x}\right)^2 + \left(\frac{\partial q}{\partial x}\right)^2 + \left(\frac{\partial p}{\partial y}\right)^2 + \left(\frac{\partial q}{\partial y}\right)^2.$$

The iterative step to compute p_{i+1} from p_i [1] is as follows:

$$p_{i+1} = \hat{p}_i + \frac{1}{\lambda} \cdot (I - R(p_i, q_i)) \cdot \frac{\partial R}{\partial p}(p_i, q_i), \quad (3)$$

where $\hat{p}_i$ is the local average of p at iteration i, I is the observed image gray level, and $R(p_i, q_i)$ is the intensity predicted by the reflectance model. A similar equation holds for q_{i+1}. The average used in this paper to compute $\hat{p}$ and $\hat{q}$ is a convolution with

$$\frac{1}{20}\begin{pmatrix} 1 & 4 & 1 \\ 4 & 0 & 4 \\ 1 & 4 & 1 \end{pmatrix}.$$

This average is slightly different from the average of four closest neighbors as originally used in [1], having somewhat better directional symmetry.

This algorithm needs boundary conditions on (p, q) in order to converge, and it can give a nonintegrable solution; integrating (p, q) on a closed path will not give zero as it should for derivatives of a true surface. Frankot and Chellappa [3] proposed to enforce integrability on given (p, q) estimates by projecting the estimates onto an appropriate space of functions using the Fourier transform. Ikeuchi and Horn [4] proposed to use Gaussian sphere coordinates instead of the (p, q) coordinates. The Gaussian sphere system enables enforcement of boundary conditions for any slope, unlike the (p, q) system that has singularity points. However, the Gaussian sphere approach was not used in this paper as the equations in the (p, q) domain are much simpler.

On a picture with N^2 pixels, one iteration step of (3) takes $O(N^2)$ operations. Including the integrability constraint of [3], additional $O(N^2 \log(N))$ operations are needed at each iteration. The total number of iterations is on the order of $O(N)$. This computational complexity suggests the introduction of a multiresolution scheme for speedup. Correct application of multiresolution turned out to be non trivial, as is shown in Section II. This correspondence will describe a method to reduce image resolution in a way which is compatible with reduction of shape resolution. Another approach to fast multiresolution shape from shading, using hierarchical basis conjugate gradient method, has recently been presented [7].

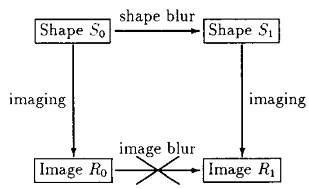

Fig. 1. Reducing image resolution and reducing shape resolution do not commute.

C. The Multiresolution Pyramid

The image pyramid, and multiresolution image analysis, have been studied extensively in the past decade [8], [2], [5], [6]. A brief review of basic pyramid techniques will be given.

An image pyramid is a sequence of copies of an original image in which resolution and sample density are decreased, usually in regular steps. Let $\boldsymbol{G}_0$ be the original image, which forms the "basis" of the pyramid. $\boldsymbol{G}_0$ is convolved with a low pass filter $\boldsymbol{w}$, then subsampled by discarding every other row and column to form $\boldsymbol{G}_1$, the "first" level of the pyramid. $\boldsymbol{G}_1$ is then filtered and subsampled to form $\boldsymbol{G}_2$, and so on. In general, for $l > 0$, this can be written concisely in terms of a convolution operator:

$$\boldsymbol{G}_l = [\boldsymbol{w} * \boldsymbol{G}_{l-1}]_{\downarrow 2}.$$

Here the notation $[\cdot]_{\downarrow 2}$ indicates the image contained within brackets is subsampled by a factor of 2 in each spatial dimension. In typical pyramids the low pass filter $\boldsymbol{w}$ is similar to a Gaussian function, and the pyramid is called a *Gaussian Pyramid.*

The filter $\boldsymbol{w}$ is called the *generating kernel.* In practice this is chosen to be small and separable, so that the computation cost of the filter convolution is kept to a minimum. Pyramid construction is a fast algorithm that generates a full set of filtered images at a cost typically less than 10 operations per pixel of the original image.

Many iterative algorithms take advantage of the pyramidal structure. The first iterations are computed on a reduced image $\boldsymbol{G}_l$ with little computational cost. The result is then expanded in size, and additional iterations are performed on $\boldsymbol{G}_{l-1}$. This process continues until the final iterations are performed on the full-size image $\boldsymbol{G}_0$. The advantage of this process is that most iterations are performed on images of reduced size, significantly decreasing the computational cost.

II. Surface Resolution and Intensity Resolution

For the intensity pyramid to be of any help in the computation of shape from shading, the following assumption should be true. Let I_0 be the gray level image of the surface S_0. Reducing the resolution of I_0 into I_1 should yield the same image as the image of surface S_1, which is obtained by reducing the resolution of surface S_0. In earlier work, Terzopoulos [9] and Simchony *et al.* [6] used a simple gray level pyramid for multiresolution shape-from-shading, with reasonable results on very simple shapes. They have used the surface computed on a reduced resolution image as an approximation for the higher resolution surface. Unfortunately, the operators of reduction of resolution and imaging (using the reflectance model) do not commute, and in general this simple approach can fail badly; see Fig. 1. Following is a one-dimensional example. In this example we will view the reduction of resolution as filtering, and we will omit the subsampling stage.

Example: Let the 1-D surface be $S_0 = \cos(2\pi x)$. Blurring S_0 into S_1 using a Gaussian filter $\frac{1}{2\pi\sigma} \cdot e^{-x^2 2\sigma}$ gives the smoother surface $S_1 = \frac{\cos(2\pi x)}{e^{2\sigma}}$. In the 1-D case, viewed and illuminated from above, (2) becomes

$$R(p) = \frac{1}{\sqrt{1+p^2}}, \tag{4}$$

where $p = \frac{\partial S}{\partial x}$.

As S_1 has smaller derivatives than S_0 at all points, its image R_1 should be brighter than the image of S_0 everywhere. And indeed S_1 is closer to a smooth plane which gives highest intensities. On the other hand, if I_1 is a smooth (blurred) version of I_0, then the relation $I_0 < I_1$ can never be true for every point. I_1 will just be closer to the average of I_0 at every point, and so $R_1 \neq I_1$. This example is also displayed in Fig. 2.

III. Blurring Surface Reflectance

In this section a method is described for estimating the reflectance R_1 of a reduced resolution surface S_1 from the gray level image $I_0 = R_0$.

It can be assumed that the blur operator and the derivative operator commute (both are convolutions). Therefore, if the surface derivatives (p,q) could be estimated from I_0 they could be blurred, and R_1 could then be generated from these smoothed derivatives. But as the main problem is to compute these surface derivatives, we will rather compute $T^2 = \tan(\alpha_{\vec{N}_s})^2$, where $\alpha_{\vec{N}_s}$ is the angle between the surface normal and the (vertical) lighting direction. From (1) and (2) we can get for every image point:

$$T^2 = \tan(\alpha_{\vec{N}_s})^2 = \frac{1-R^2}{R^2} = p^2 + q^2, \tag{5}$$

and

$$T = \left|\tan(\alpha_{\vec{N}_s})\right| = \sqrt{\frac{1-R^2}{R^2}} = \sqrt{p^2+q^2}. \tag{6}$$

T is computed from the image intensities R. Given T, R can be computed by using the following equation:

$$R = \sqrt{\frac{1}{1+T^2}}. \tag{7}$$

As $T = \|(p,q)\|$ is the norm of (p,q), then $\alpha T = \|(\alpha p, \alpha q)\|$. Simply stated, multiplying T by a constant corresponds to multiplying the surface derivatives by a constant. This is used to estimate the function T of the reduced resolution surface given the original surface. Resolution reduction involves blur followed by subsampling, and the critical stage is the blur. Therefore, the function T of the blurred surface should be estimated given the function T of the original surface.

When the derivatives (p_1,q_1) and (p_2,q_2) are similar at two adjacent point, then $\|(p_1,q_1)+(p_2,q_2)\|$, the norm of the smoothed surface, can be approximated by $\|(p_1,q_1)\| + \|(p_2,q_2)\|$, smoothing the two surface norms. This holds, of course, for any local convolution operator. So blurring the surface derivatives (p,q) by convolving with a Gaussian, and then computing T of the blurred surface, can be approximated by blurring the original T. This approximation is more accurate when the derivatives (p,q) do not change quickly, i.e., the surface is smooth. If on the other hand the derivatives (p,q) do change quickly, then the estimate will no longer be accurate. Blurring T, the norm of (p,q), will give higher values than the norm of $(blurred(p), blurred(q))$. But even in this case the image obtained after blurring T will be closer to the correct image than a simple gray level blur.

In summary, to estimate the image of the blurred surface we take the original image R, compute from it the array T using (6), blur T, and recompute an intensity image using (7). This procedure gives the exact image of the low resolution surface for one dimensional case where there is no sign changes in the slope. For other cases this procedure is an approximation giving a much closer estimate than just smoothing the image. This is displayed clearly in Fig. 3, where the reduced resolution images are getting brighter as the slopes are getting smaller.

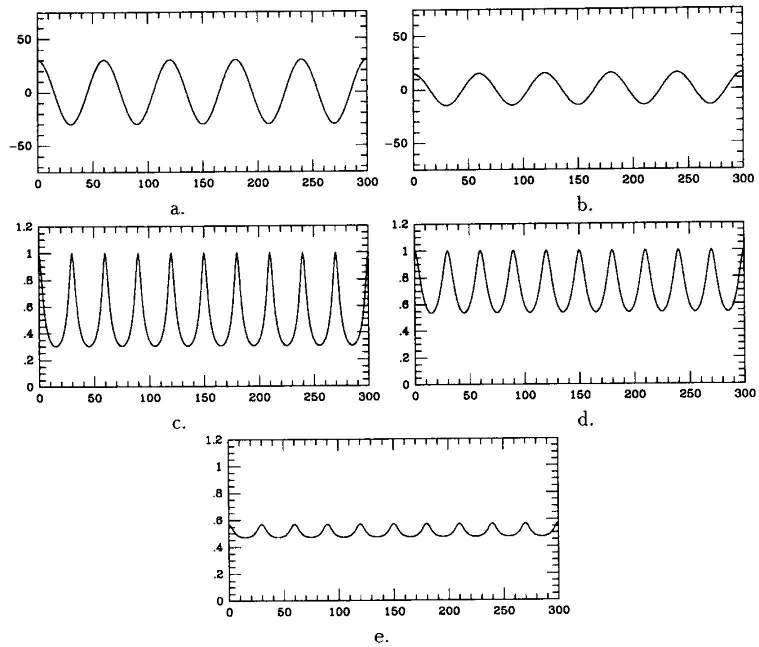

Fig. 2. Blurring the surface and blurring the image do not commute. (a) A surface $S_0 = \cos(\alpha)$. (b) The surface S_0 of (a) blurred to give $S_1 = \frac{1}{2}\cos(\alpha)$. (c) The intensity $R(S_0)$ of the original surface using (4). (d) The intensity $R(S_1)$ of the blurred surface in (b). (e) Blurring the intensity in (c). This blurred intensity is different from the intensity of the blurred surface in (d).

IV. Pyramidal Scheme for Shape-from-Shading

Based on the value T as computed in (6), the suggested algorithm for building the gray level pyramid for shape from shading purposes is as follows:

1) T_0 is calculated from the given input image I_0 using (6).
2) A Gaussian pyramid $T_0 \cdots T_{n-1}$ is being built, whose basis is T_0, as described in Section I-C. The pyramid has n levels, chosen such that T_{n-1} will still have significant shape information. We have experimented with pyramids whose smallest levels were of size 32×32.
3) Using (7), the gray level image R_i is calculated for every pyramid level T_i. The images R_i constitute a gray level pyramid, which is the estimate of the reduced resolution surface reflectance.
4) The multiresolution shape-from-shading algorithm is performed using the R_i pyramid. First, (p_{n-1}, q_{n-1}) are computed from R_{n-1} using any shape-from-shading algorithm, for example the one described in Section I-B. The obtained low-resolution derivatives (p_i, q_i) are than expanded into (p_{i-1}, q_{i-1}), to be used as an initial guess for the computation of the (p, q) derivatives at the level $i - 1$ of the pyramid. The process is repeated until the (p, q) derivatives are computed for the full resolution image.

With this pyramidal scheme most iterations are being performed on low resolution images, saving significant computations.

V. Experimental Results

The proposed multiresolution scheme has been tested on two simulated surfaces of size 128×128, displayed in Fig. 3. I_0 was numerically calculated from the surfaces using (2). A three-level reflectance pyramid was then built using the scheme of the previous section: $R_0 = I_0$ of size 128×128, R_1 of size 64×64, and R_2 of size 32×32. The blurring of the T-pyramid was done

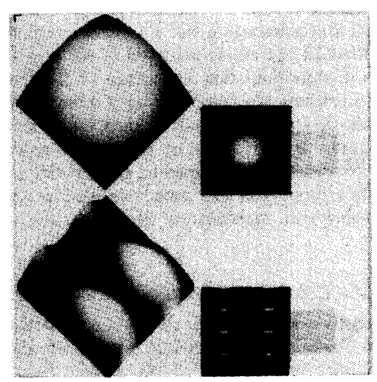

Fig. 3. The reflectance pyramid. (a) A perspective view of the original surfaces, size 128×128. (b) The intensity images of the original surfaces. (c), (d) Reducted resolution intensity images to sizes 64×64 and 32×32. The resolution is reduced according to the algorithm described in Section IV.

using the following mask:

$$\begin{pmatrix} 0 & 0.125 & 0 \\ 0.125 & 0.5 & 0.125 \\ 0 & 0.125 & 0 \end{pmatrix}.$$

The surfaces, as well as the intensity images of the reduced surfaces, are displayed in Fig. 3. It should be noted that the overall brightness of the reduced resolution images is significantly higher than the brightness of the higher resolution images. This occurs as the reduced resolution surfaces are more flat, their brightness should thus increase, and this effect is captured in our scheme for resolution reduction.

Boundary conditions for (p, q) where calculated from S_0 on the surface boundaries, and reduced in resolution for the other levels. The coefficient λ of (3) was taken to be 7000 in all the runs.

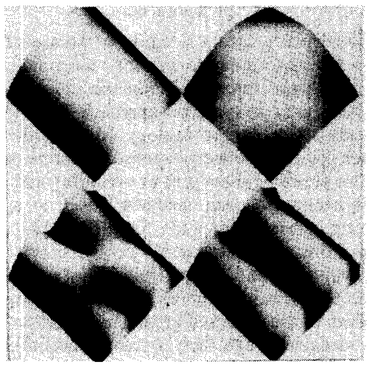

Fig. 4. Perspective view of the surfaces obtained using nonpyramid algorithms. Left: Without using the integrability constraint (*sfs*). Right: Using the integrability constraint at each iteration (*sfsi*).

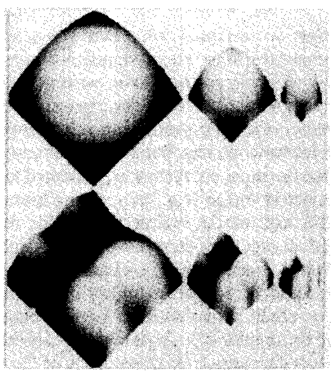

Fig. 5. Perspective view of the surfaces obtained in the multiresolution approach of Section IV, without using the integrability constraint. The lowest resolution was computed first, and then the higher resolution surfaces.

Fig. 4 shows the surfaces reached by shape from shading algorithms without using pyramids. It shows both the regular shape from shading without using the integrability constraint (*sfs*), and using the integrability constraint (*sfsi*). In the case which did not use the integrability constraint during the iterations, it was applied once after convergence to enable the creation of a displayable surface.

Figs. 5 and 6 display the surfaces obtained using the pyramidal algorithm with and without the integrability constraint. Fig. 5 displays the surfaces in the multiresolution pyramid at all three levels.

Fig. 6 displays the surfaces in the multiresolution pyramid when the integrability constraint was enforced after every iteration.

All algorithms gave results of similar quality, differing mainly in the computational cost. Table I summarizes the number of iterations needed to reach convergence, and the error between the calculated and the known (p, q). The computational speedup of the multiresolution approach is evident, and this speedup improves with the complexity of the images.

VI. Concluding Remarks

A method has been presented for image resolution reduction which simulates the reduction of shape resolution, to be used with shape-from-shading algorithms. Resolution reduction has been used for accelerating shape from shading algorithms using multiresolution pyramids.

The goal of this correspondence is to show that resolution reduction based only on image intensities can be inferior to resolution reduction using knowledge of the surface reflectance. This is true for all cases of parameter estimation from intensity images, when the intensity is not a linear function of the parameters.

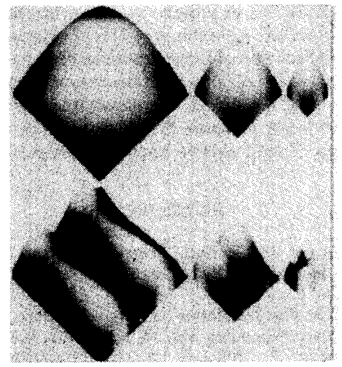

Fig. 6. Same as Fig. 5, but with applying the integrability constraint at every iteration.

TABLE I

Summary of the Number of Iterations and the Final Error with all Tested Methods. The Multiresolution Methods Include the Number of Iterations at Every Resolution Level. The Upper Table Represents Results for the Simpler Shape, While the Lower Table Represents Results for the More Complicated Shape. The Methods Are Named as Follows: *sfs*: Shape-from-Shading. *msfs*: Multiresolution Shape-from-Shading. *sfsi*: Shape-from-Shading with the Integrability Constraint. *msfsi*: Multiresolution Shape from Shading with the Integrability Constraint.

Results for the Simpler Shape:

Algorithm	Size	Iterations	$E^2(p)$	$E^2(q)$
sfs	128 × 128	70	0.08	0.08
msfs	32 × 32	13		
	64 × 64	4		
	128 × 128	6	0.136	0.136
sfsi	128 × 128	13	1.006	1.006
msfsi	32 × 32	11		
	64 × 64	6		
	128 × 128	5	0.495	0.495

Results for the Complicated Shape:

Algorithm	Size	Iterations	$E^2(p)$	$E^2(q)$
sfs	128 × 128	81	6.3	1.7
msfs	32 × 32	18		
	64 × 64	6		
	128 × 128	15	8.9	3.11
sfsi	128 × 128	54	3.5	10.3
msfsi	32 × 32	8		
	64 × 64	7		
	128 × 128	8	3.11	5.17

Since optical reduction of images, as done when moving a camera away from the object, is a simple gray level averaging, a question is cast on the general scheme of shape-from-shading. As we have shown that in general reduction of gray level resolution does not correspond to scale reduction of the object, it follows that for accurate shape-from-shading we need to know the distance from the object, and actually the shape should also be known in advance.

References

[1] M. J. Brooks and B. K. P. Horn, "Shape and source from shading," in *Int. Joint Conf. Artificial Intelligence,* Los Angeles, CA, Aug. 1985, pp. 932–936.

[2] P. J. Burt, "Fast filter transforms for image processing," *Comput. Vision, Graphics, Image Processing,* vol. 16, pp. 20–51, 1981.

[3] R. T. Frankot and R. Chellappa, "A method for enforcing integrability in shape from shading algorithms," *IEEE Trans. Pattern Anal. Machine Intell.,* vol. 10, pp. 439–451, July 1988.

[4] K. Ikeuchi and B. K. P. Horn, "Numerical shape from shading and occluding boundaries," *Artificial Intell.,* vol. 17, pp. 141–184, 1981.

[5] A. Rosenfeld, Ed., *Multiresolution Image Processing and Analysis.* New York: Springer-Verlag, 1984.

[6] T. Simchony, R. Chellappa, and Z. Lichtenstein, "Pyramid implementation of optimal step conjugate search algorithm for some computer vision problems," in *Proc. Second Int. Conf. Computer Vision,* Tampa, FL, Dec. 1988, pp. 580–590.

[7] R. Szeliski, "Fast shape from shading," in *Proc. First European Conf. Computer Vision,* Antibes, France, Apr. 1990. New York: Springer-Verlag, pp. 359–368.

[8] S. L. Tanimoto and T. Pavlidis, "A hierarchical data structure for picture processing," *Comput. Graphics Image Processing,* vol. 4, pp. 104–109, 1975.

[9] D. Terzopoulos, "Efficient multiresolution for computing lightness, shape from shading, and optical flow," in *Proc. Fourth Nat. Conf. Artificial Intelligence (AAAI-84),* 1984, pp. 314–317.

A New Set of Constraint-Free Character Recognition Grammars

L. Stringa

***Abstract*—A general pattern recognition procedure is presented for application to unconstrained alphanumeric characters. The procedure is designed to allow hierarchical redescription of the input images in terms of significant elements only, and formal developments are given within the framework of elementary phrase-structure grammars. The extraction of the primitives associated with the terminal vocabularies of the grammars is always deterministic, and the productions of the parsers are characterized by a significant degree of topological fidelity. Preliminary experimental results indicate recognition rates comparable to the state of the art, but with a considerable reduction in computing time.**

***Index Terms*—Constraint-free character recognition, hierarchical classification, implementability, parsing, pattern compression, pattern primitives, phrase-structure grammar.**

Manuscript received February 9, 1989; revised March 7, 1990. Recommended for acceptance by C. Y. Suen.

The author is with the Istituto per la Ricerca Scientifica e Tecnologica, 38050 Povo (Trento), Italy.

IEEE Log Number 9037570.

I. Introduction

This correspondence presents a family of phrase-structure grammars designed for application to pattern recognition, specifically, printed, typed, or handwritten alphanumeric characters. Following the so-called structural or syntactic approach [12]–[14], [25], the proposed schema relies on the analogy between the *hierarchical* structure of such patterns and the *syntactical* structure of a language's expressions. This seems to allow greater versatility and independence of quantitative parameters than methods based on a statistical or decision-theoretic approach [4], [15], [21], [28], [39]: each pattern is analyzed in terms of its structural "components" using a small set of pattern primitives and composition rules, and the recognition procedure takes the form of a simple linguistic parsing.

A major feature of the proposed grammars is that the extraction of the primitives associated with their terminal vocabularies is always deterministic. Except for rough vertical normalization and centering, no specific constraint on the input data (such as standardized writing within preprinted frames) is required.

II. Outline of the Approach

The construction is patterned with two goals in mind: on the one hand, the procedure must attain significant independence from a variety of styles, sizes, imperfections in writing, noise, etc.; on the other hand, it must guarantee a high description rate, i.e., keep to a minimum the number of ambiguities tolerated. Otherwise, the solution might have some intrinsic interest, but would lack any practical utility.

In the following, the input characters are assumed to be given in the form of a binary matrix image (e.g., by entering them into an optical scanner). Since the size of the matrix reflects the resolution of the digitized character, it can vary depending on the degree of accuracy one wants to achieve. For example, a character size of 8 (wide) by 10 (high) pixels could be sufficient for stylized type fonts such as those imprinted on credit cards, whereas slightly bigger matrix sizes might be required for regular type fonts such as Courier, Elite, and the like; for handwritten characters, the dimensions of the matrix could be even bigger, as the number of relevant features increases considerably [35]. For the moment, no restriction is assumed on the matrix size, provided that it meets some obvious readability conditions. Thus, for example, the pictures in Fig. 1 may count as input instances of the character "2" (as produced by two distinct handwritings) digitized in matrices of sizes 16×32 and 8×14, respectively.

Due to the enormous variety of writing styles and graphical variations thereof, a "correct" interpretation of such images would obviously involve a huge amount of data processing unless the number of reference patterns can be reduced. For this reason, a procedure will be defined to allow a redescription of the input images (regardless of their size) in terms of significant elements only, thereby facilitating the recognition process. Also, raw data may carry a certain amount of noise as well as redundant information (isolated pixels, breaks, gaps, bumps, etc.). To eliminate such elements, it is common practice to introduce, at the input end, a preprocessor that "smoothes out" the digitized characters in accordance with certain fixed standards. Such techniques are often very complicated, but seem to be necessary if the patterns are analyzed in terms of particular geometric features or line segments (such as bars, hooks, arches, loops, and the like), which could become aleatory due to the practical difficulties involved in the extraction procedure [3], [8], [10], [11], [17], [22]–[24], [34], [38]. The primitives used here will instead be quite simple and the extraction procedure totally deterministic. This simplicity permits the more uniform and compact treatment required by the first goal expressed above.

The proposed procedure can be described as follows. Given an input image I_0, it is first divided into a certain number of subimages, which will be assumed to be square windows of side $h \geq 2$. To each of these windows, a new square (or primitive) of side $j \leq h$