Range Image Acquisition with a Single Binary-Encoded Light Pattern

P. VUYLSTEKE AND A. OOSTERLINCK

Abstract—This paper addresses the problem of strike identification in range image acquisition systems based on triangulation with periodically structured illumination. The existing solutions to strike identification rely on subsequent projection of multiple encoded illumination patterns. These coding techniques are restricted to static scenes. We present a novel coding scheme based on a single fixed binary encoded illumination pattern, which contains all the information required to identify the individual strikes visible in the camera image. Every sample point indicated by the light pattern, is made identifiable by means of a binary signature, which is locally shared among its closest neighbors. The applied code is derived from pseudonoise sequences, and it is optimized as to make the identification fault-tolerant to the largest extent. This instantaneous illumination encoding scheme is suited for moving scenes.

We realized a working prototype measurement system based on this coding principle. The measurement setup is very simple, and it involves no moving parts. It consists of a commercial slide projector and a CCD-camera. A minicomputer system is currently used for easy development and evaluation. Experimental results obtained with the measurement system are also presented. The accuracy of the range measurement was found to be excellent. The method is quite insensitive to reflectance variation, geometric distortion due to oblique surface inclination, and blur of the projected pattern. The major limitation of this illumination coding scheme was felt to be its susceptibility to the effects of high contrast textured surfaces.

Index Terms—Depth maps, error-correcting codes, pseudonoise sequences, range-finding, space-encoding, structured light, 3-D vision, triangulation.

I. INTRODUCTION

TRIANGULATION with structured light is a well established technique for acquiring range pictures of three-dimensional scenes. The spatial position of a point is uniquely determined as the intersection of a straight line with a plane, of three planes, or as the intersection of two straight lines within a plane. These geometric principles have been applied in wide range of simple triangulation techniques, using a directional illumination device (a projector), and a directional sensor (a TV-camera). The basic schemes use a directed light beam to indicate a single object point [1]–[3], or a light sheet to indicate a cross-section of the visible scene surface [4]–[6]. An additional

deflection mechanism with one or two degrees of freedom makes it possible to cover the full field of view. Various techniques have been developed to improve the accuracy [7], to expand the range of depth [8] or the field of view [9], or to realize a compact design [10], [11].

These triangulation methods with a simple illumination pattern yield unambiguous measurement results, but the acquisition speed is rather low, optomechanical deflection devices tend to be delicate, and the available sensor surface is sparsely used. One could consider to project a fan of light sheets instead of a single one in order raise the acquisition speed [12], but this extension is not obvious since the light stripes are not individually recognizable in the camera image, and consequently it is not known from which light sheet they originate. This identification problem is illustrated in Fig. 1; it is similar to the well-known correspondence problem in stereo vision [13]. If a point grid (or an orthogonal line grid) is projected instead of a parallel line grid, then the ambiguity may be alleviated by taking into account the discrete nature of the projected grid [14], [15], or it may even be eliminated by applying the epipolar constraint several times in a dynamic configuration where the projector is successively rotated [16]. The latter technique requires that the observed grid points be tracked from frame to frame.

The identification problem has been solved in a direct way by *encoding* the illumination pattern. The "intensity ratio" method [17] is based on monotonic modulation of light intensity across the scene. An additional reference illumination is required to cancel the effect of varying reflectance. Tajima used a single monochromatic illumination pattern with gradually increasing wavelength (i.e., a rainbow pattern), in combination with a two-channel color camera [18]. Both techniques provide for high lateral resolution (limited by the number of camera pixels), but acceptable range accuracy can only be obtained with a camera combining excellent response stability with very large dynamic range. Furthermore it is not obvious how to deal with the effects of mutual illumination from nearby reflecting surfaces.

These inconveniences are greatly reduced by implementing the illumination pattern with only two intensity levels, "bright" and "dark." It is still possible to provide for sufficient discrimination between say $2^n - 1$ stripes with only two intensity levels by projecting n binary patterns *in succession*. This means that every point to be identified is marked with a sequence of n bits adding
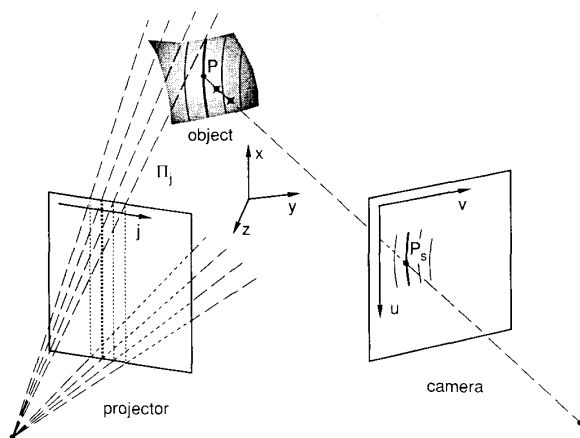
Fig. 1. The identification problem with multiple stripe triangulation. An observed image point P_s on a light stripe may correspond to any of the projected light sheets Π_j. The index value j is required to compute the coordinates of the actual object point P.

up to a unique codeword. A few alternative schemes of this time-sequential encoding technique have been described in literature, using an encoded point grid instead of a line grid [19]; a Hamming [20] or Gray code [21] instead of a simple binary code. The effects of reflectance variation may be canceled by making use of additional reference illumination patterns (full bright, full dark), or even a set of complementary patterns [22]. Very accurate alignment of the subsequent pattern masks is generally obtained with a programmable liquid crystal shutter window. These range acquisition systems seem to perform excellently, but they are not suited for moving scenes due to their time-sequential nature. For any movement between two subsequent coding steps would result in erroneous identification.

It is therefore essential to restrict the encoding process to a *single* instance (e.g., by means of a flash light source), such that all the information required for identifying the indicated scene points is available in a single camera picture. In the scheme proposed by Boyer and Kak [23] a single colored stripe pattern is projected on the scene. Only very few different colors are used, typically three or four. The illumination pattern can be thought of as a concatenation of identifiable subpatterns (typically six stripes wide). The recognition of any valid subpattern codeword in the observed stripe sequence leads to the identification of the stripes in that subpattern, and possibly also of some adjacent stripes (by some kind of "crystal growing" process).

Our approach which we shall present next shows some similarity to the latter, in that the indexing information is *locally* distributed. However while the encoding scheme of Boyer and Kak is based on *adjacent* subpatterns, assigned to fixed positions in the stripe pattern, we developed a code grid in which identifiable subpatterns *overlap*. This means that any window of some fixed minimal size in an arbitrary position of the illumination grid will

return a codeword which uniquely identifies the column index. Our option is motivated by the inherent lack of synchronization, i.e., the boundary between adjacent subpatterns is not explicitly available in the fixed subpattern approach. The conceptual difference between adjacent and overlapping subpatterns is also very important in the perspective of fault-tolerance: in [23] the color sequence is assigned to the line grid in such a way that there is a specified minimal code distance between every pair of (fixed boundary) subpatterns. However, since the decoder is not informed about the subpattern boundaries, real fault-tolerance may only be guaranteed if one specifies a minimal code distance between every pair of windows at arbitrary grid column positions. These considerations have led to a novel binary illumination-encoding scheme, which we shall present next. First we focus on the basic principles of our encoding technique, and the resulting illumination pattern layout. The third section deals with the various aspects of an experimental measurement system based on this encoding technique. Experimental results are presented in the last section.

II. Illumination-Encoding Based on Pseudonoise Sequences

We have investigated a space-encoding scheme based on the following target specifications. 1) A square grid of points is projected on the scene. 2) The projected grid points must be easily detectable in the camera image, and it should be possible to accurately estimate their image coordinates (u, v) (cf. Fig. 1). 3) The index value j of an observed grid point must be identifiable (i.e., the column number of the grid points that are projected on the scene by the plane Π_j). In a calibrated measurement setup the knowledge of the image coordinates (u, v) and the index j is sufficient to compute the spatial position of the corresponding object point. 4) All this information must be supplied by means of a single binary illumination pattern.

With binary modulation it is very hard to associate more than a few bits with every grid point to make it identifiable. One could consider some kind of shape-modulation, which alters the height, width, orientation, compactness, or the like to discriminate between individual grid points. It should be noted however that most of these features are not invariant with respect to perspective distortion. Moreover they are very susceptible to noise if the grid spacing is very small (only a few pixels in the camera image). We decided therefore to assign only a *single codebit* to every grid point. With this low information density it is still possible to identify the corresponding index value, by systematically distributing the binary information among neighboring grid points in such a way that every window of adjacent grid points of some specified size yields a codeword which uniquely identifies the index of the grid point. This means that the grid points are identifiable only within their *local context*. Thus for valid identification it is required that the original context of a point in the projected grid be preserved locally in the camera image. This assumption holds true almost everywhere in the image,

except at jump boundaries, where the projected bit map will be locally disturbed. Any identification window which contains such a context disruption must not be used for index identification.

Ideally any context disruption should be detectable by the fact that the observed window value is inconsistent, i.e., that it is not compatible with the projected bit map. An optimally designed binary code grid should therefore comply with the following requirement: a specified degree of fault-tolerance must be guaranteed with the smallest possible identification window size. The degree of fault-tolerance may be expressed as the minimal number of binary positions in which any two windows (taken from arbitrary columns in the grid) differ, i.e., the minimal Hamming distance between any window pair. We developed a highly redundant code grid based on pseudonoise sequences, the details of which will be elaborated in Section II-B. First, we shall present a modulation scheme suited for the implementation of such a binary illumination code.

A. Basic Layout of the Illumination Pattern

An appropriate illumination pattern with only two intensity levels which 1) indicates a grid of points and 2) marks each grid point individually with a single codebit, is depicted in Fig. 2. Note the basic chess-board appearance. By definition, the grid points to be indicated on the scene are located at the intersection of the horizontal and vertical chess-board edges, i.e., where four squares meet. Upon projection every column generates a plane Π_j, the index j of which must be known for locating the indicated object points by triangulation. In our experimental measurement system the entire pattern consists of 63 columns of 64 points each (only 10 rows are shown).

For the purpose of identification every grid point is individually marked with a bright or a dark spot, representing a binary "1" or "0", respectively. Additionally the chess-board naturally exhibits two basic appearances of the grid points, which alternate both in horizontal and vertical direction. We denote them as either "+" or "−". Hence one may think of the complete pattern to be composed of the four primitives depicted in Fig. 3.

The assignment of individual codebits to the grid points is based on two binary sequences $\{c_k\}$ and $\{b_k\}$, of 63 bits length. The grid points on the kth column of the "+" type are assigned the binary value c_k, while those of the "−" type are assigned b_k, as illustrated in Fig. 4. As a consequence the bit map assignment is repeated every second row. With a proper choice of the sequences $\{c_k\}$ and $\{b_k\}$ every window of 2 × 3 grid points at any arbitrary grid position carries a 6-bit codeword (composed of an ordered c_k-triplet and a b_k-triplet) which is in one-to-one correspondence with the underlying column number or index $j = k$. This means that an identification window size of 2 × 3 is sufficient for unique identification. The actual choice of the sequence pair is very critical with respect to the uniqueness of the identification—if a larger window is used it will also influence the degree of fault-
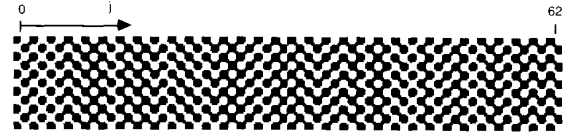

Fig. 2. Detail of the projected binary pattern.

0+   0−   1+   1−

Fig. 3. The entire illumination pattern is composed of four primitives.

Fig. 4. Two binary sequences are bit-wise assigned to the grid such that every window of 2 × 3 bits has a unique signature in every position when it is shifted horizontally. Points of the "+" type are marked with a bit of the $\{c_k\}$ sequence, and those of the "−" type with a bit of $\{b_k\}$. The assignment is repeated every second row.

tolerance—but this topic is postponed till the next subsection. At present we simply assume that such sequences exist.

We now concentrate on the main characteristics of the illumination pattern. The choice of a chess-board as the basic structure has been motivated by some typical advantages. For a specified grid density its bandwidth is only half that of a point grid (or a parallel line grid, if only one direction is considered). The grid points in the chess-board are defined at the intersection of the horizontal and vertical edges. In general it is very difficult to estimate the exact location of an edge in an image without scene-dependent bias, since the observed intensity profile across an edge may vary according to the local surface inclination and reflectance. In the case of a chess-board, however, the horizontal position of a grid point is defined by two vertical edge segments (meeting at the grid point) which have opposite sign. In the same way two horizontal edge segments with opposite sign define the vertical position of the interjacent grid point. If one assumes the surface characteristics to be locally smooth, then the observed edge locations will be biased in opposite sense by equal amounts, both for the horizontal and vertical edge pairs, so these offsets will tend to cancel out in estimating the location of the grid points. Notice that for triangulation the precision of the range measurement critically depends on the accuracy of the grid point coordinates in the camera image.

The specific choice of the additional modulation which superimposes a bright or a dark spot at every grid position has been motivated by the following arguments. First, deciding whether a spot is bright or dark in the camera image is largely facilitated by the fact that both a bright and dark reference is *locally* supplied at any grid position. If the evaluation of the spot brightness (and hence the codebit value) is made relative to some function (e.g., the mean) of these reference intensities, then the effects of reflectance variation and ambient light may be canceled out, as long as these parameters vary smoothly *within* the reference neighborhood. Second, the actual image positions where the codebit primitives (i.e., the spots) have to be evaluated are clearly indicated since the latter coincide with the grid vertices. A third argument concerns the interference between the basic modulation component for locating the grid points and the additional component for marking them with a codebit. It is clear that the observed gray-level distribution in the immediate neighborhood of a grid point will depend on the actual value of the corresponding codebit. However, the spot being bright or dark will not bias the estimation of the grid point location, since the gray-level function is locally point-symmetric with respect to the grid point in either case. This is the main reason why we rejected some other binary modulation schemes in which the identification primitives did *not* coincide with the grid vertices.

The proposed identification scheme based on local binary signatures requires that the network of grid points be locally reconstructed in the camera image. The chessboard structure facilitates the search of adjacent grid points in horizontal and vertical direction, since every pair of 4-neighbors is connected by a high contrast edge segment. Without this pictorial information 4-neighbors and diagonal neighbors might be confused in regions where the observed pattern is highly distorted. In fact it is sufficient to consider that immediate horizontal or vertical neighbors should have opposite "sign" (in the sense mentioned earlier).

The code grid has been designed such that any grid point may be identified by the signature of a 2×3 window covering its immediate neighbors. Consequently, any pair of adjacent rows of the grid carry different code information. In general, if the identification should be based on a $t \times b$ window format, then the bit map must be organized such that any t consecutive rows provide for independent information. This requirement may be achieved by repeating the bit assignment every t rows. In the simplest case when the bit map is composed of equal rows ($t = 1$) the appropriate identification window format would be very elongated (covering at least 6 adjacent grid points in horizontal direction, if 63 column positions have to be identified). We mentioned however that a window will fail to be identified if it is disrupted by a jump boundary. Obviously a compact window is less likely to be disrupted than an elongated one. This is the main reason why we preferred a (2×3)-identification scheme instead of (1×6).

Although the latter approach of distributing the identification data among t subsequent rows leads to a more compact window format, at the same time it introduces additional ambiguity. At any given column position an identification window may exhibit t different signatures, depending on its row position. In fact shifting a window over the bit map in vertical direction causes its rows to rotate. Without special precautions the same signature may be found at different column positions, if both windows do not belong to the same row. This problem is illustrated in Fig. 5, with $t = 2$. Although the Hamming distance between any pair of windows on the same row is at least 2, they cannot be uniquely identified unless their actual row index (modulo 2) is known. One could subject the bit map design to additional constraints, such that a minimal Hamming distance between every pair of windows is guaranteed, irrespective of their relative row positions, but this would increase the window size for a specified minimal distance. The ambiguity is elegantly solved for $t = 2$ by taking into account the alternate appearances of vertices in the chess-board (indicated as "$+$" or "$-$"). With the specific assignment as in Fig. 4 the c_k-bits in a window may always be distinguished from the b_k-bits irrespective of the vertical window position, since they are assigned to grid points of opposite sign. In other words the additional sign information makes it possible to reorder the contents of any arbitrary window to a standard six-tuple, in which c_k- and b_k-bits are at fixed positions. This way both windows indicated in Fig. 4 would correspond to the same 6-tuple ((c_3, c_4, c_5), (b_3, b_4, b_5)).

B. Bit Map Assignment

For any arbitrary identification window format one may always define an equivalent n-tuple with a specified order of c_k- and b_k-bits, n being the total number of bits in the window. The window even need not be rectangular. For some specified window format the corresponding n-tuple may be regarded as a codeword, and the set of all the codewords which are obtained by considering the window at every horizontal position on the bit map defines a code. Every window format defines a distinct code. If the identification is to be performed with a window of some specified format, say $2 \times b$ bits, then a pair of binary sequences $\{c_k\}$ and $\{b_k\}$ should be found such that the corresponding code is optimal, in the sense that it has the largest possible minimal Hamming distance. Finding an optimal code of rather small dimension in general is not a problem, since the best ones have been exhaustively described in the literature [24]. However, the codes to be found in our specific case are subject to a strong additional restriction, due to the fact that the identification windows overlap. This implies that the codewords cannot be chosen independently.

We investigated the application of a special kind of recursive sequences, called pseudonoise sequences, to generate an optimal bit map. A binary sequence generated

011`1111`010101100110111101`101`0010011100010111100101000110000100000
101`1101`10100100111000010`1111`00101000110000100000111111010101011001
0111111010101100110111101`101`0010011100010111100101000110000100000

Fig. 5. Ambiguity may arise from the fact that the grid code is composed of more than one sequence: the indicated windows belong to different columns, although they carry the same signature.

by the recursion (modulo 2):

$$c_k = \sum_{i=1}^{m} - h_i c_{k-i}, \qquad k \geq m \qquad (1)$$

with initial values $c_0, c_1, \cdots, c_{m-1}$, is an mth order PN-sequence if the corresponding characteristic polynomial defined as:

$$h(x) = \sum_{k=0}^{m} h_k x^k, \qquad h_0 = 1, h_m = 1 \qquad (2)$$

is an irreducible factor of $x^k - 1$ for $k = 2^m - 1$, but not for any smaller k, which means that it is a primitive polynomial. If the initial values are not all zero then the resulting sequence will have period $p = 2^m - 1$, which is the maximal period length achievable with a linear recurrence. For more details on PN-sequences we refer to [25] and [26]. Two properties deserve particular attention. By the "window" property every group of m consecutive bits is unique over an entire period of the sequence. This is clearly the required property for unique identification. Secondly the set consisting of the p-bit words obtained by cyclically shifting a given period of a PN-sequence, extended with the all-zero word is a cyclic code having dimension m, word length $p = 2^m - 1$, and a constant Hamming distance $d = 2^{m-1}$ [27], which means that any pair of words exactly differ by one more bit than the number of positions in which they do agree.[1]

Now let $\{c_k\}$ be an mth order PN-sequence, and $\{b_k\}$ = $\{c_{k-\varphi}\}$ the same sequence with a phase difference φ, and let these sequences be assigned to the bit map as illustrated in Fig. 4. Then the code defined by applying a window to every position of the bit map is equivalent to the code which results from puncturing the simplex code defined by the PN-sequence. This means that the resulting code is obtained by systematically deleting specified bit positions in the original simplex code (for more information on puncturation see [28]). Every window format corresponds to a specific puncturation.[2] If the window size is $2 \times b$ then in the punctured code all bit positions will be deleted except for two groups of b bits wide, separated by $\varphi - b$ positions. This is illustrated in Fig. 6.

Starting from a simplex code with Hamming distance $d = 2^{m-1}$, it is clear that the minimal distance will either diminish by one, or will be unaffected, every time a bit position is deleted. In the special case when $\varphi = b$, the

---

[1]This kind of equidistant cyclic code is called a simplex or minimal code.

[2]In fact, the moving window may not generate every code word of the punctured simplex code, unless the bit map is cyclically extended at its left and right borders, such that the window is defined over the entire horizontal range of p positions. The all-zero word is also missing.

$\varphi = 5$

		simplex code	punctured code
$\{c_k\}$	→	011110101100100	011010
		111101011001000	111101
		111010110010001	111011
		110101100100011	110110
		101011001000111	101100
$\{b_k\}$	→	010110010001111	010001
		101100100011110	101010
		...	...

Fig. 6. A simplex code of dimension 4 ($n = 15$), and the resulting punctured code of length $n = 6$, when a 2×3 window is applied to a grid which is composed of the first and the sixth word of the simplex code (i.e., two periods of the corresponding PN-sequence with a relative phase shift $\varphi = 5$).

punctured code will consist of $2b$ adjacent positions of the original simplex code. This situation is equivalent to applying a row-window of $2b$ bits wide to a bit map consisting of equal rows defined by a single PN-sequence. For a given PN-sequence and window size, φ is the only parameter which influences puncturation. Consequently the minimal code distance resulting from an optimal twin-row organization will be at least as large as the distance corresponding to a single-row scheme, since in the former approach puncturation may be carried out with an additional degree of freedom.

We examined the distance behavior of sixth-order PN-sequences ($m = 6; p = 63$) and the influence of the relative phase shift φ. In a twin-row organization unique identification will require a window size of at least 2×3. Hence our first criterion states that $d_6 = 1$, where d_{2b} denotes the minimal Hamming distance obtained with a $2 \times b$ window. Where possible the decoding scheme will use a larger window size, so that local signature inconsistencies may be detected. Therefore we also maximize a second criterion in order to optimize the bit map simultaneously for a set of larger window sizes, more specifically for the range between 2×4 and 2×8:

$$\sum_{b=4}^{8} \frac{d_{2b} + m - 1}{2 \times b} \text{ is maximal.} \qquad (3)$$

This criterion is derived from the Singleton distance bound, which states that any code of word length n, dimension m, and distance d (i.e., an $[n, m, d]$-code) must satisfy $(d + m - 1)/n \leq 1$. In applications like this where the number of essentially different PN-sequences (only three of sixth order) and the number of meaningful phase combinations are rather small, it is feasible to find the optimal sequence combination by exhaustive search. According to the above criteria one sixth-order sequence-phase combination was found to score better than all the other ones, i.e., the PN-sequence with characteristic polynomial $h(x) = 1 + x + x^6$, combined with the same sequence shifted by $\varphi = 17$ positions:

$$\{c_k\} = 1111110000010000110001010011110100011100010010110111011001101010$$

$$\{b_k\} = \{c_{k-17}\}. \qquad (4)$$

TABLE I
COMPARISON OF THE CODES OBTAINED BY APPLYING WINDOWS OF SIZES
$(2 \times 3)-(2 \times 8)$ TO THE BIT MAP GENERATED BY THE SEQUENCES (4),
WITH THE BEST CODES DESCRIBED IN LITERATURE

window size $2 \times b$	code distance d_{2b}	$(d_{2b}+5)/2b$	best code of equal dimension m=6, and equal length n=2b		$(d+5)/n$
2×3	1	1	[6,6,1]		1
2×4	2	0.875	[8,6,2]		0.875
2×5	3	0.8	[10,6,3]	short. Hamm.	0.8
2×6	4	0.75	[12,6,4]		0.75
2×7	4	0.643	[14,6,5]	short. BCH	0.714
2×8	5	0.625	[16,6,6]		0.688

Moreover, a better performance was obtained in comparison with the single-row scheme over the equivalent range of window sizes (8-16 in 2-increments), for any of the three sequences. This demonstrates that an identification scheme based on rectangular windows, besides yielding more compact windows, also may result in a higher degree of fault-tolerance. The codes generated by applying the windows of sizes 2×3 through 2×8 to the bit map specified by the optimal sequence pair (4) are compared in Table I with the best codes described in literature [24]. Although these codes simultaneously result from a single bit map, they prove to be optimal up to the window size 2×6, in the sense that no known codes of equal word length and dimension have a larger Hamming distance. We also investigated the use of nonlinear recursive sequences and combinations of different PN-sequences, but their distance behavior was found to be worse. More details are supplied in [29].

III. EXPERIMENTAL RANGE MEASUREMENT SYSTEM

We realized a prototype based on this binary encoding principle. The measurement setup is very simple and it involves no moving parts. The basic geometry is sketched in Fig. 1. The illumination pattern depicted in Fig. 2 has been recorded on a 24×36 mm Ortho-negative and is projected on the scene with a standard slide projector equipped with a 250 W xenon bulb and a 90 mm/f1:2.5 objective. The entire pattern consists of 64 rows by 63 columns. The camera (High Technology Holland, MX type) has a MOS-CCD frame transfer image sensor of 604(H) $\times$ 576(V) pixels. It delivers a 625-line CCIR video signal (linear response, constant gain), which is sampled and digitized to a 512×512 by 8 bit digital picture with 1:1 aspect ratio. The focal length of the camera objective (16 mm/f1:1.8) has been matched to the projection optics, such that approximately the entire illumination pattern is visible when it is projected on a frontal surface at nominal depth. In that case the observed grid period roughly corresponds to 8 pixels in both directions.

The digitized picture is processed by a VAX-750 minicomputer. Demodulating and decoding an illumination-encoded image comprise the following steps: 1) detecting the grid points, estimating their precise location in the image, determining their appearance ("sign") in the chess-board, and their individual codebit indicated by a bright or dark spot; 2) searching for the closest N-, S-, W-, and eastern neighbors of every detected grid point, i.e., recovering the local grid context in the image; and 3) identifying grid points by their local signature, using dynamic windows of some specified minimal size, and checking for signature consistency. Identified points are next spatially located by triangulation. We developed a program package which also includes routines for geometric calibration, and diagnostic and display routines for the presentation of intermediate and final results. The programs are written in Pascal. The flexibility of a high level environment with many I/O-facilities makes the developed software package a powerful tool for evaluating the performance and limitations of the new range acquisition method.

The processing time is quite long (50 s for a range image of 64×63 points), but 80 percent of this time is spent to low-level picture processing, which is well suited for hardware implementation. The size of the complete program package is of the order of 10 000 lines of (commented) source code; the specific routines for processing the encoded image and for triangulation comprise only some 1000 lines. 1.3 MB of memory are needed for data storage, including the input image. In the following subsections we outline the relevant characteristics of the subsequent processing steps. A more extensive algorithmic description is presented in [29]. For the purpose of illustration consider the picture of Fig. 7 which represents a scene consisting of a vase, a teapot and a cup illuminated by the coded pattern. We denote this single input picture by the discrete function $g(u, v), 0 \leq g(u, v) \leq 255, 0 \leq u \leq 511, 0 \leq v \leq 511$.

A. Demodulation

The detection of grid points in the image is based on correlation with the reference pattern:

$$1/9 \begin{bmatrix} -1 & -1 & -1 & 0 & 1 & 1 & 1 \\ -1 & -1 & -1 & 0 & 1 & 1 & 1 \\ -1 & -1 & -1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \end{bmatrix}$$

$$\equiv 1/9 \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} * \begin{bmatrix} -1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 \end{bmatrix}$$
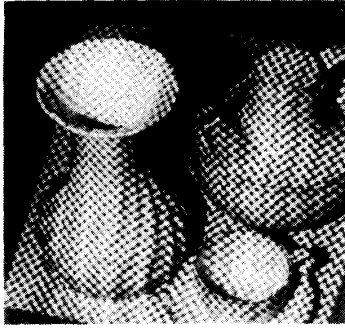
$$(5)$$

Fig. 7. Illumination-encoded picture of a vase, a teapot, and a cup. This digitized picture is the only input required for computing the corresponding range image.

which is actually implemented as a 3×3 local averaging operation followed by correlation with a 5×5 sparse reference pattern. If we denote the local average image by $\bar{g}(u, v)$, then the correlation image is obtained by performing the local operation:

$$c_0 = \bar{g}_{-2,2} + \bar{g}_{2,-2} - \bar{g}_{-2,-2} - \bar{g}_{2,2} \qquad (6)$$

where the subscripts indicate the relative pixel offsets with respect to the reference point of the considered neighborhood. This intrinsic image will exhibit extrema at the grid points, the negative peaks corresponding to grid points of the "−" type and the positive peaks to the "+" type grid points.

Every pixel which meets the following requirements is retained as a grid point:

$$|c_0| \text{ is a local maximum AND} \qquad (7)$$

$$|c_0| > T_1 \left(|\bar{g}_{-2,2} - \bar{g}_{2,-2}| + |\bar{g}_{-2,-2} - \bar{g}_{2,2}| \right) \text{ AND}$$
$$\qquad (8)$$

$$|c_0| > T_0 \qquad (9)$$

where the local maxima are considered within centered 7×7 neighborhoods, and T_0 and T_1 are threshold parameters. The heuristic term (8) is unaffected by offset or gain changes in the input image, since these effects cancel out at both sides of the inequality. The adaptive threshold expression at the right-hand side of (8) also improves selectivity, for it represents the degree of asymmetry of the picture function along both diagonals through the candidate grid point; it will approach zero if the opposite bright and dark squares are equally bright, which is a typical characteristic of a grid point observed without severe distortion. The net effect of this criterion is illustrated in Fig. 8(a). For comparison, Fig. 8(b) shows the response of pure correlation. A rather low fixed threshold T_0 in the last term (9) is applied additionally for eliminating false matches in image regions of very low contrast.

The precise location of a grid point may be estimated by fitting a (e.g., quadratic) surface through the correlation data points in a small neighborhood around the corresponding local maximum, and computing the analytical maximum of the fitted surface. We followed an alternative approach based on the center of mass of the correlation peaks, which requires fewer computations. In fact the
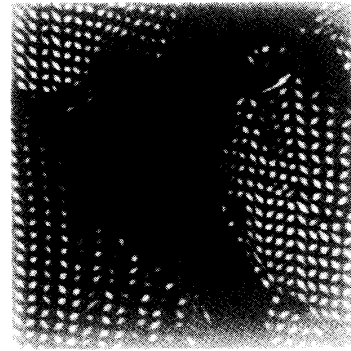
(a)
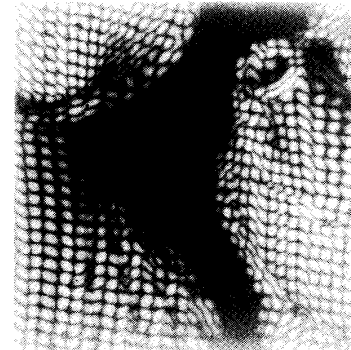
(b)

Fig. 8. Correlation images of the vase-teapot-cup scene (200×200 pixel details taken from the center). (a) Response of the operation $|c_0| - (|\bar{g}_{-2,2} - \bar{g}_{2,-2}| + |\bar{g}_{-2,-2} - \bar{g}_{2,2}|)$, with $T_1 = 1$ and all negative values clipped to black. (b) Absolute value of pure correlation $|c(u, v)|$.

center of mass is a good approximation to the correlation maximum, and even coincides with it if the correlation function is symmetric with respect to its maximum. Since the projected pattern shows the required symmetry the latter condition will generally be met unless the object surface exhibits nonlinear reflectance variation in the immediate neighborhood of the considered grid point, e.g., when the surface is strongly textured. If (u_0, v_0) represent the pixel coordinates of the local correlation maximum, then the subpixel offsets of the grid point coordinates are estimated as:

$$\delta u_0 = \frac{\sum_{x=-1}^{1} \sum_{y=-1}^{1} t(c(u_0 + x, v_0 + y))x}{\sum_{x=-1}^{1} \sum_{y=-1}^{1} t(c(u_0 + x, v_0 + y))}$$

$$\delta v_0 = \frac{\sum_{x=-1}^{1} \sum_{y=-1}^{1} t(c(u_0 + x, v_0 + y))y}{\sum_{x=-1}^{1} \sum_{y=-1}^{1} t(c(u_0 + x, v_0 + y))} \qquad (10)$$

with $t(c) \equiv c - Bc(u_0, v_0) \qquad$ if $c - Bc(u_0, v_0) > 0$

$$\equiv 0 \qquad \qquad \text{otherwise.}$$

The weight coefficients $t(c)$ are computed by referring the correlation values to some basic level, which is specified as the Bth fraction of the local maximum, and clip-

ping negative values to zero. This means that only the top of the correlation peak is considered in computing the center of mass. Optimal results were obtained with $B = 0.75$–0.80.

We estimated the precision of this subpixel location technique by means of the following experiment. If the illumination pattern is projected on a flat target then ideally every set of grid points belonging to the same row or column should be collinear in the camera image. In our test procedure collinearity is examined by fitting lines to the rows and columns of located grid points in the image, according to the least squared error criterion. The overall RMS deviation of the grid points with respect to these lines is then used as a rough measure for the random location error. Many types of flat test scenes were used, including a white frontal plane, frontal planes with different kinds of printed texture such as stripes and text, and tilted surfaces. The RMS deviation ranged from 0.11 pixels (H and V) for both the white plane and the striped plane, to 0.28 (H) and 0.19 (V) pixels for a target plane tilted by $60°$ (parallel with the camera-projector base line). For comparison, we also estimated the grid point locations by means of least squares quadratic surface fit around the local correlation maxima, using a 3×3 neighborhood of correlation values just like in the center of mass method. Nearly identical results were obtained, but at the expense of additional computation time. Although the above figures do not represent the actual location errors (which might include systematic components) they clearly indicate the feasibility of our subpixel location approach, since 0.30 pixels RMS deviation was the best result obtained without subpixel estimation.

The final step in demodulation consists of determining the "sign" of every located grid point, which is simply the sign of the correlation value at that point; and determining its codebit, i.e., deciding whether the superimposed spot is bright or dark. It is clear that simply thresholding the image function at the spot locations is not appropriate since reflectance and the amount of ambient light may vary considerably across the scene. Determining the codebits requires a well-balanced adaptive criterion, which makes up for smooth brightness fluctuations. In the case of a chess-board pattern the observed bright and dark levels tend to be equally distributed around any grid point, so the corresponding local gray-level histogram will be bimodal and symmetric. Since bright and dark spots have equal *a priori* probability, the local mean gray-level of a grid point neighborhood provides an optimal threshold for deciding whether the spot at the center is bright or dark. More specifically the threshold level at a grid point is computed as:

$$f_b = 1/8 \left(\sum_{k=-1}^{1} \sum_{l=-1}^{1} \bar{g}_{3k,3l} - \bar{g}_{0,0} \right) \qquad (11)$$

which corresponds to the mean of a 9×9 neighborhood of the input picture, with exception of the 3×3 central pixels. The brightness of the spot is evaluated by a weighted sum of $\bar{g}$-pixel values in a 3×3 neighborhood, the weight coefficients being determined by the actual subpixel offset of the grid point. This adaptive criterion
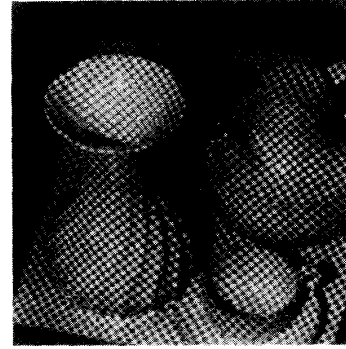

Fig. 9. Results of demodulation for the vase-teapot-cup scene. The locations of detected grid points are indicated by either horizontal or vertical bars of 3 pixels long, superimposed on the input image. The direction of the bars denotes the value of the corresponding code-bit, "|" ≡ 1, "−" ≡ 0, respectively. The grid points (located up to 1/4 pixel) are displayed at the nearest integer pixel positions, due to the limited display resolution.

proves very robust unless the symmetric distribution assumption is heavily violated, either by the presence of high contrast texture, or by nonlinear effects like saturation of the camera response. The results of demodulation are illustrated in Fig. 9 for the vase-teapot-cup scene.

B. Decoding

The previous steps served to characterize every detected grid point by its image coordinates, its "sign," and its corresponding codebit. For the purpose of identification, however, windows of adjacent grid points are to be considered instead of isolated grid points. The actual configuration of codebits in a window determines a signature by which the enclosed grid points can be identified. Hence the first decoding step consists of locally recovering the original grid topology in the observed image, i.e., finding the natural N-, S-, W-, and E-neighbors of any detected grid point. A directed graph, with a node for every detected grid point, and branches to the immediate 4-neighbors, is very convenient to represent these relational data.

Finding neighboring grid points may be facilitated by invoking some heuristics, which follow from geometrical considerations. The geometry of our experimental system (depicted in Fig. 1) is subject to some weak restrictions, which we denote by a "usual" triangulation setup: 1) the projector and the camera point to some common point so as to optimally cover the working space; 2) their mutual separation (base length) is small compared to the nominal object distance; 3) the focal length/image size ratios of the projector and the camera are nearly equal and relatively large; and 4) the j-axis does not deviate very much from the base line direction (i.e., the line through both lens centers). Under these circumstances the grid rows will be observed as nearly parallel and equidistant in the camera image. In other words, the direction and vertical separation of the grid rows is only to a small extent modulated by the scene geometry. Hence the search area for finding neighbors in the image may be confined to parallelogram-shaped regions at fixed relative distances, as
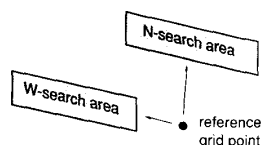
Fig. 10. Finding the N- and W-neighbors of any grid point is confined to parameterized regions at fixed offsets.
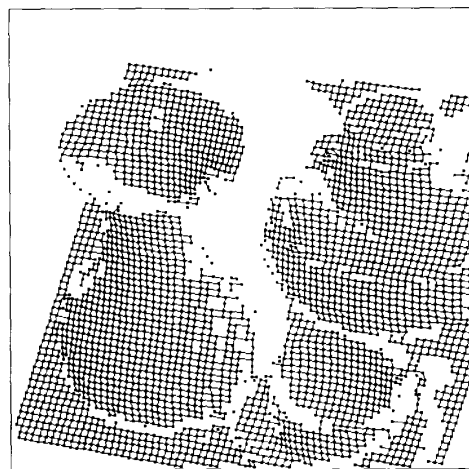

Fig. 11. Result of establishing links between direct neighbors. Although this information is displayed pictorially as physical segments connecting the grid points, it is actually stored by means of a graph data structure, the nodes of which represent the located grid points.

illustrated in Fig. 10. The actual width, height, and orientation which specify these search areas, and their position relative to the reference grid point, are derived from statistical data gathered during system calibration.

Pairs of direct neighbors (both in horizontal and vertical direction) are then selected according to the least Euclidean distance criterion, taking into account that two candidate neighbors must have opposite "sign," and one of them lies in the others search area. For every detected neighbor pair a (bidirectional) link is created which connects the corresponding nodes in the graph. The results of local grid reconstruction are displayed in Fig. 11. After the linking process, all essential information is available in a graph data structure, in which every node includes the following data fields: the image coordinates (u, v) of the corresponding grid point (up to 1/4 pixel), its "sign" and codebit, pointers to the N-, S-, W-, and E-neighbor, and corresponding flags for indicating which neighbors are missing.[3]

The identification is implemented as a two-stage process. First a window of codebits is iteratively evaluated at every grid point. Starting at the grid point to be identified a signature window is grown essentially in the W- and E-directions, by repeated reference to the respective pointers in the graph. If the horizontal expansion gets locked at some stage because the W- or E-neighbor is missing, then an attempt is made to proceed by rerouting the search via a single step in the N- or S-direction; e.g., if the W-neighbor is not available, then the W-search proceeds along the NW-neighbor instead, or if the latter is also missing, along the SW-neighbor. The expansion is equally continued in both horizontal directions, unless either of the search paths is exhausted, in which case the search proceeds only in the other direction. At every stage of expansion one or two codebits are appended two the current signature: one associated with the considered grid point, and the other one drawn from its immediate N- or S-neighbor (if any). These bits, which correspond to a tuple (c_k, b_k) of the PN-sequences which build up the grid

[3]For the purpose of execution speed this graph structure is actually being created while searching for the local correlation maxima. Both the pixel image data structure and the graph are used until after the grid links have been established. Moreover the graph is organized as an array of 128 × 128 nodes, such that every fourth pixel (in both directions) of the image array corresponds to a node. It is thereby explicitly assumed that every neighborhood of 4 × 4 pixels contains at most one grid point. Each node is provided with an additional flag, indicating whether it effectively corresponds to a grid point, or is empty. This organization of the graph, like a low resolution image array, makes it very straightforward to access the appropriate node starting from an image pixel. The reverse way is effectuated by recording the image coordinates of a grid point (if any) in the appropriate field of the corresponding node.

code, are the only information available along the current grid column, since they alternate in vertical direction (Fig. 4). They are mutually distinguishable by the sign of the corresponding grid point. The actual identification of the grid point under consideration is accomplished by iterative elimination. Initially every element of the integer set $\{0 \ldots 62\}$ is considered to be a valid index number. The knowledge of the binary code tuple at the start position already eliminates 3/4 (or 1/2 if only one codebit is available) of the initial candidate index numbers. At every subsequent stage of the iterative window expansion the acquaintance of an extra code tuple further reduces the set of valid index numbers. We implemented this elimination scheme as follows: the set of valid index numbers for some grid point at any iteration stage is represented by a 63-bit vector, which is initially set to all ones, indicating that any index is valid. The elimination is accomplished by repeated AND-ing the latter with a vector of the same dimension, representing the set of index values (indicated by ones at the corresponding positions) that are compatible with the observed code tuple. A lookup table is very convenient for generating these elimination vectors, since the latter are entirely specified by the acquired code tuple, and the current horizontal offset within the search window. The elimination procedure is terminated if: 1) a specified minimal number of codebits N_b have been acquired, or 2) the search gets locked in both directions, or 3) a specified maximal offset in either the W- or E-direction has been exceeded. After termination valid identification is obtained if N_b codebits could be acquired and if the resulting index vector is all-zero except for one position, which then indicates the actual index number of the grid point. An all-zero index vector points to the fact that the local signature is inconsistent. The error-detection performance is directly determined by the specified minimal window size N_b.

Most of the grid points are identified with this iterative

procedure. Near jump boundaries however many grid points may not be identified due to the limited expansion possibilities, so that the iteration procedure is terminated before N_b codebits could be acquired. In a second stage an attempt is made to identify these grid points by some kind of index propagation process. If the index numbers of the existing 4-neighbors of a nonidentified grid point are mutually compatible, then the latter is identified accordingly, at least if its (possibly incomplete) code-tuple is compatible with the proposed index value. This propagation process is repeated a few times. Fig. 12(a) shows the result of identification before propagation, and (b) after three propagation iterations.

C. Triangulation

We modeled the camera according to the "pinhole" equations given by:

$$[tu \ tv \ t]^T = S[x \ y \ z \ 1]^T \quad (12)$$

in homogeneous coordinates, where the (3×4)-matrix S is called the camera matrix. These equations specify the mapping of an object point in Cartesian coordinates (x, y, z) with respect to an arbitrary world frame, to the corresponding image point coordinates (u, v), expressed in pixel units (cf. Fig. 1). These equations include the transformation from world to camera coordinates, perspective projection, rescaling, and misalignment corrections. Although the projector can be modeled exactly the same way, we preferred to use explicitly the equations of each of the 63 planes Π_j through the grid columns and the lens center. These are of the form:

$$x + l_2^{(j)}y + l_3^{(j)}z + l_4^{(j)} = 0, \quad j = 0 \dots 62. \quad (13)$$

The first coefficient $l_1^{(j)}$ is set to one, so for every plane Π_j only three independent parameters need to be determined. These equations are suited to represent any plane that is not parallel to the x-axis. One can rewrite the equations (12) and (13) as a set of three linear equations in the unknowns x, y, z, and solve them numerically, given the image coordinates of a point (u, v), and the corresponding index j. It is however possible to solve this set of equations symbolically, so the spatial solution can be expressed as a linear transform of the image coordinates:

$$[hx \ hy \ hz \ h]^T = M^{(j)}[u \ v \ 1]^T \quad (14)$$

where the (4×3)-matrix $M^{(j)}$ (the so-called sensor matrix) is computed in terms of the camera matrix S and the coefficients $l_i^{(j)}$, $i = 1 \dots 4$, for each of the 63 planes Π_j. This direct implementation of triangulation, adopted from Bolles, Kremers, and Cain [30], is much faster than the conventional technique of numerically solving the set of equations (12) and (13).

D. Calibration

Computing the components of the sensor matrices $M^{(j)}$ for a given measurement setup requires the knowledge of the camera matrix S and the coefficients $l_i^{(j)}$ of
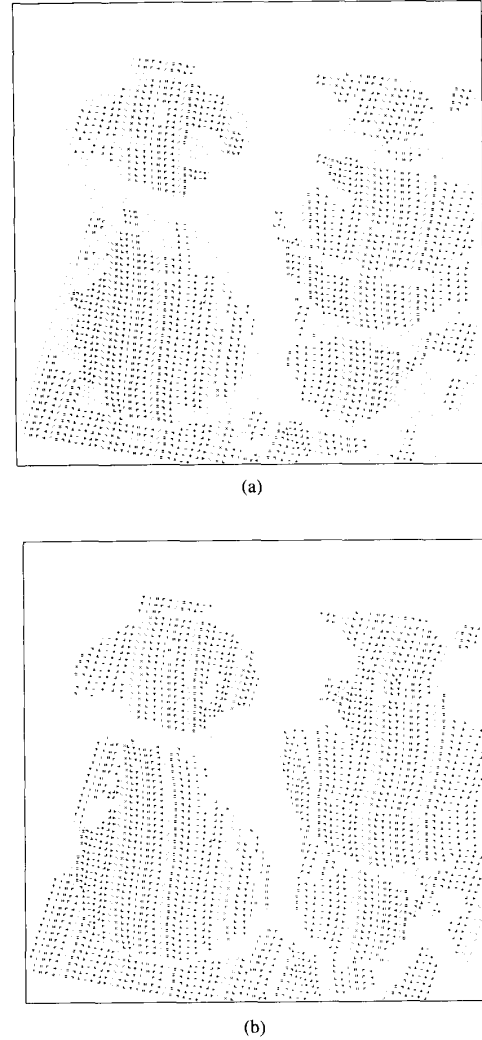

(a)

(b)

Fig. 12. Result of identification, (a) before index propagation, with a specified minimal window size $N_b = 9$, and (b) after three propagation iterations. The identified index values are represented by graphical symbols. Only 16 different symbols were available, hence they indicate the index modulo 16. Detected grid points which could not be identified are represented by a dot.

the planes Π_j. A standard technique for computing the components of the camera matrix, using a set of calibration points with known world coordinates, has been reported by Rogers and Adams [31]. Given a set of N calibration points with observed image coordinates (u_i, v_i), and known world coordinates (x_i, y_i, z_i), the camera equations (12) can be rewritten as a set of $2N$ nonhomogeneous linear equations, involving the 11 components of the camera matrix as unknowns (the lower right component of S being arbitrarily set to one). For obtaining a well-conditioned set of equations, the focal length must not be very large, and the calibration points (at least six) have to be nicely spread over the space of view. The effect of measurement noise is reduced significantly if a very

large number of calibration points are being involved, so the camera parameters are computed as the least squares solution of a largely overconstrained set of equations. In practice however the use of many calibration points (a few hundred) is rendered difficult due to the requirement that each of them must be identifiable in the camera image. We solved this problem by making the calibration points identifiable in exactly the same way as the grid points in the encoded illumination pattern. The calibration setup consists of a flat panel, parallel to the x-y-plane, which can be fixed at calibrated z-positions (cf. Fig. 1). The panel surface is uniformly covered with a set of identifiable points at calibrated (x, y)-positions. Hence for some specified position ($z = 0$), the calibration panel actually *defines* the world coordinate system. A set of evenly distributed calibration points is obtained by placing the panel at a few calibrated positions across the focus range of the camera, and taking a picture in each case. Fig. 13 shows the layout of the calibration panel, which is essentially the same as the projected pattern used for triangulation, but in which only 63 isolated windows of 2 × 7 grid points are made visible on a white background. By definition the upper middle grid point of every window corresponds to a calibration point, the (x, y)-coordinates of which have to be accurately measured in advance. Each calibration point is identified by the 14-bit signature of the surrounding window. The identification is quite robust since the Hamming distance between any arbitrary pair of windows is at least 4. Locating and identifying the calibration points in the subsequent images is accomplished with essentially the same routines as those used for demodulating and decoding illumination-encoded images. Moreover this calibration scheme makes it possible to consider a large number of calibration points (up to 252, using 4 images), with minimal human intervention.

The second part of calibration consists of determining the coefficients $l_i^{(j)}$ of the planes Π_j. At this point we assume that the camera matrix has already been calibrated. A large set of calibration points is generated by projecting the encoded illumination pattern on a white panel, subsequently placed at a few calibrated z-positions within the focus range of both camera and projector, using the same mechanical equipment as in the case of camera calibration. The world coordinates of these calibration points are uniquely solved from the inverse camera equations (12), since the known constant z-value of the panel surface can be used as additional constraint. These calibration points are next partitioned into 63 subsets of points with equal index number, so every subset will consist of all calibration points belonging to the same plane Π_j. Finally the plane coefficients $l_i^{(j)}$ are computed by fitting planes through each subset of calibration points. The points within a subset are not collinear, since they are distributed among the intersection lines of the plane Π_j with the calibration panel, placed at subsequent depth positions. With 4 panel positions, up to 256 calibration points are used to compute the corresponding least squares plane, which ensures excellent noise immunity.
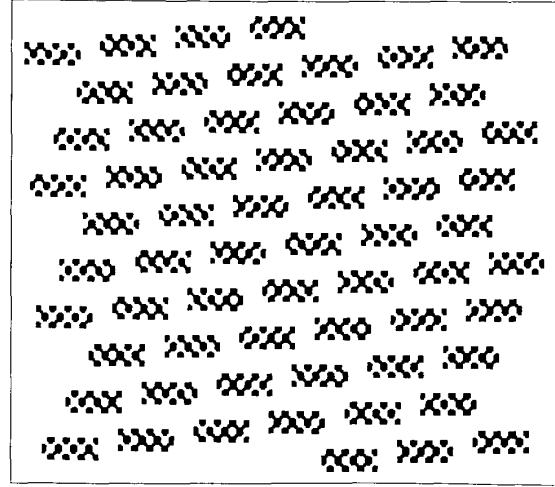

Fig. 13. Layout of the calibration panel. The 63 windows are at known calibrated positions on the panel, and are mutually distinguishable.

IV. Experimental Results

A series of elementary experiments have been carried out to estimate the measurement accuracy and the performance of the prototype under varying circumstances, including dynamic depth range, surface reflectance and orientation range, and the extent to which textured surfaces do interfere with it. The same functional parameter values have been used in the experiments described next: $T_0 = 36$; $T_1 = 1$; $B = 0.75$ (cf. (9), (8), and (10), respectively); the identification is accomplished with a minimal window size $N_b = 9$, followed by three index propagation iterations. These parameter values proved appropriate throughout our experiments. Moreover we found that small deviations from these default values do not significantly affect the measurement performance (except for N_b: when this parameter is enlarged fewer points will be identified, but the identification error rate will also decrease, and conversely; $N_b = 9$ or 10 yielded the maximum number of correctly identified points in most experiments).

A. Calibration

We verified the appropriateness of the linear camera model by backprojecting the imaged calibration points (used for camera calibration) on the planes of constant z from which they originate (i.e., the calibrated panel positions). For panel positions ranging from 95–140 cm w.r.t. the camera-projector base line, the RMS-deviation in the panel plane between the actual calibration points and the backprojected points varied from 0.1–0.2 mm. These figures are indicative of the small nonlinear lens distortion, and the high precision of the grid point location (as far as random errors are concerned).

The range precision has been evaluated by measuring points on the (white) calibration panel with the fully calibrated system, and considering the distance offset of the resulting range points with respect to the known panel positions. In the case of a triangulation setup with a camera-

projector base width of 39 cm, the RMS depth error was found to be 0.3 mm; this measurement comprised some 14 200 points lying in four planes, at distances between 105–140 cm from the base line. Although the absolute range precision might be worse, this measure still includes the error components caused by nonlinear behavior (lens aberrations, grid distortion, panel curvature), and noise in locating the grid points, which has been our major concern.

B. Range Limitations

Making use of a flat test scene we were able to evaluate the measurement system quantitatively under extreme circumstances. In each of these experiments a least squares plane is fitted through the obtained range points. Two basic parameters are considered to be indicative of the measurement performance: the RMS value of the distance d of the range points with respect to the fitted plane, and the total number of range points N_{cop} lying within a distance to the plane of less than 2 mm. The first measure concerns the measurement precision, while the second figure, i.e., the number of coplanar points, is almost equal to the number of grid points that have been correctly identified, since any misidentification is likely to result in a large range error.[4] With N_{det} representing the number of detected grid points in the image, and N_{id} the number of grid points that have been identified (not necessarily correctly), we also consider the following ratios to be significant: N_{cop}/N_{det}, which is the relative number of correctly identified points, and $(N_{id} - N_{cop})/N_{id}$, representing the error rate of identified points.

In our experimental measurement system the dynamic depth range is mainly limited by the focus depth of the projector, due to the large (fixed) f-number 1 : 2.5. Although the projected pattern already shows noticeable blur when the target is displaced only by a few centimeters from the plane of focus, the range measurement showed no remarkable degradation within the depth range from 90–115 cm. Numerical results (using a white frontal target plane) are presented in Table II. Although N_{det} is smaller than 4032 outside the range 100–105 cm, we verified that all *visible* points have been effectively detected in these experiments. The fact that many points are missing outside this range is exclusively due to the limited field of view of the camera. The same experiments were repeated with the projector lens aperture reduced by a factor of 4 (and the camera aperture enlarged accordingly). At 80 cm 73 percent of the points were correctly identified, 99 percent at 85 cm, 87 percent at 125 cm, and 84 percent at 130 cm. In the range 85–125 cm the identification error rate was less than 0.9 percent.

The object reflectance range within which the system

[4]For that reason only points within a distance of 2 mm to the plane are considered in computing the RMS deviation. Also the actual reference plane is determined iteratively, in such a way that a new estimate of the plane is based on only the 50 percent most coplanar points obtained from the previous iteration. This eliminates the influence of fatal errors on the fitted reference plane.

TABLE II
DEPTH RANGE OF THE MEASUREMENT SYSTEM AT 1 m NOMINAL DISTANCE. THE NUMBER OF DETECTED, IDENTIFIED, AND CORRECTLY IDENTIFIED (I.E., COPLANAR) GRID POINTS ARE TABULATED AS A FUNCTION OF TARGET DISTANCE (W.R.T. THE BASELINE), USING A FLAT FRONTAL TARGET. THE LAST COLUMN SHOWS THE RMS DISTANCE OF THE RANGE POINTS TO THE FITTED REFERENCE PLANE

Depth (cm)	N_{det}	N_{id}	N_{cop}	N_{cop}/N_{det}	$(N_{id}-N_{cop})/N_{id}$	RMS d (mm)
85	1438	776	-	-	-	-
90	3310	3214	3191	0.96	0.007	0.2
95	3712	3708	3708	1.00	0.000	0.2
100	4032	4031	4031	1.00	0.000	0.2
105	4032	4031	4031	1.00	0.000	0.3
110	3776	3757	3757	0.99	0.000	0.3
115	3520	3291	3291	0.93	0.000	0.4
120	3300	2507	2434	0.74	0.029	0.6
125	2841	1922	1502	0.53	0.219	0.7

keeps on working (with fixed lens aperture) has been evaluated by means of a series of gray target planes, with relative reflectance ρ varying from 15 to 100 percent, the latter value corresponding to standard white paper. All points were correctly identified within the reflectance range 32–100 percent. At ρ equal to 22 percent, 97 percent of the points were identified, with an error rate of 0.3 percent. In this case the mean gray-level of the input image was only 16 units (of 256), and the RMS value 17. The latter figures refer to smooth reflectance variation. We also evaluated the system behavior with respect to textured surfaces, making use of different textures on a white (frontal) background. These included text patterns with characters ranging from 3.5 to 14 mm (at 0.8 m distance), and vertical line patterns (2.5 mm pitch and 50 percent duty cycle) with contrast ratio from 2 : 1 to 8 : 1. The results are shown in Table III. All grid points were visible in the camera image. The considerable susceptibility to the effect of high contrast textured surfaces is clearly one of the major limitations of this coding technique.

As long as the object surface is nearly perpendicular to the bisector line of both optical axes (which we denote by "frontal" position), and the angle subtended by the axes is not too large, geometric distortion of the observed grid will be very small. Rotating the target about its vertical axis by some angle θ away from the frontal plane causes the observed horizontal pitch to decrease or increase, depending on the sign of θ. Tilting the object by ϕ about an axis parallel to the base line causes the observed grid columns to deviate from the vertical direction. We examined the maximal range of θ and ϕ within which the system keeps functioning correctly, using a flat target. The results for the extreme target positions are shown in Fig. 14. In the first case, Fig. 14(a), when the target plane was nearly oblique w.r.t. the camera, $\theta = -65°$, the mean horizontal pitch was 4.1 pixels. The other extreme, $\theta = 65°$, Fig. 14(b), corresponds to a mean horizontal pitch of 12.9 pixels. In Fig. 14(c) the plane was tilted by 70°, causing the observed grid columns to decline by 33°. The RMS distance d of the range points to the fitted reference plane was: (a) 0.2 mm, (b) 0.3 mm, and (c) 0.7 mm,

TABLE III
INFLUENCE OF TEXTURE ON MEASUREMENT PERFORMANCE. THE FRONTAL
TARGET PLANE IS AT 0.8 m FROM THE BASELINE

Scene		N_{det}	N_{id}	N_{cop}	$N_{cop}/4032$	$(N_{id}-N_{cop})/N_{id}$	RMS d (mm)
(1) text 3.5mm		4032	3309	3190	0.79	0.036	0.3
(2) text 7mm		4037	3530	3429	0.85	0.029	0.4
(3) text 14mm		4003	3437	3333	0.83	0.030	0.4
(4) lines 2.5mm, contrast 8:1		3397	1637	1445	0.36	0.117	0.5
(5)	4:1	3963	1919	1697	0.42	0.116	0.4
(6)	2:1	4032	3210	3073	0.76	0.043	0.3

respectively (at 156 cm nominal distance). We do not mention any quantitative results on the identification performance, because it was not clear how many grid points actually were visible in the image. Nevertheless, the pictures plainly reveal the high detection rate, and negligible identification error rate [only two points in Fig. 14(b)].

C. Real Scenes

After triangulation every identified grid point is spatially located by its world coordinates (x, y, z). These spatial data are difficult to interpret however. Hence for easy visualization the 3-D grid points are subjected to a perspective transformation (the parameters of which are adjusted interactively, in order to get a "nice" view). A wire frame representation of the range image is obtained by connecting neighboring grid points in the perspective view.[5] The results displayed in Figs. 15–19 have been acquired with a setup in which the camera and the projector were tilted down by approximately 40°, and their optical axes subtended an angle of about 14°. Fig. 15 shows two wire frame views of the range points acquired from the vase-teapot-cup scene. The axes of the superimposed world coordinate frame [clearly visible in Fig. 15(b)], are 10 cm long in reality. Fair results have been obtained with a scene consisting of a multicolor printed box and spray cans, as shown in Fig. 16. A camera picture of the scene without encoded illumination (b) has been added for the sake of clarity. Only the large text on the smallest side of the box significantly interferes with the range acquisition. Fig. 17 illustrates the effect of spatial texture. The surface envelope of the long-haired brush and the surface of the bricks are properly reproduced. Although the lateral resolution of the range acquisition is inadequate to pick up details of the keyboard in Fig. 18, only a few identification errors have been encountered. The adverse influence of specular reflection is apparent in Fig. 19, where points of the overexposed corner region of an untreated coachwork part are missing in the wire frame. Fig. 20 illustrates the suitability of this range acquisition method for measuring the human body. An application in detecting spinal deformity is currently being investigated. The impracticability of fixing the body during the measurement requires that the acquisition be carried out *instantaneously*. This motivates the use of a nonsequential encoding technique like the one described here.

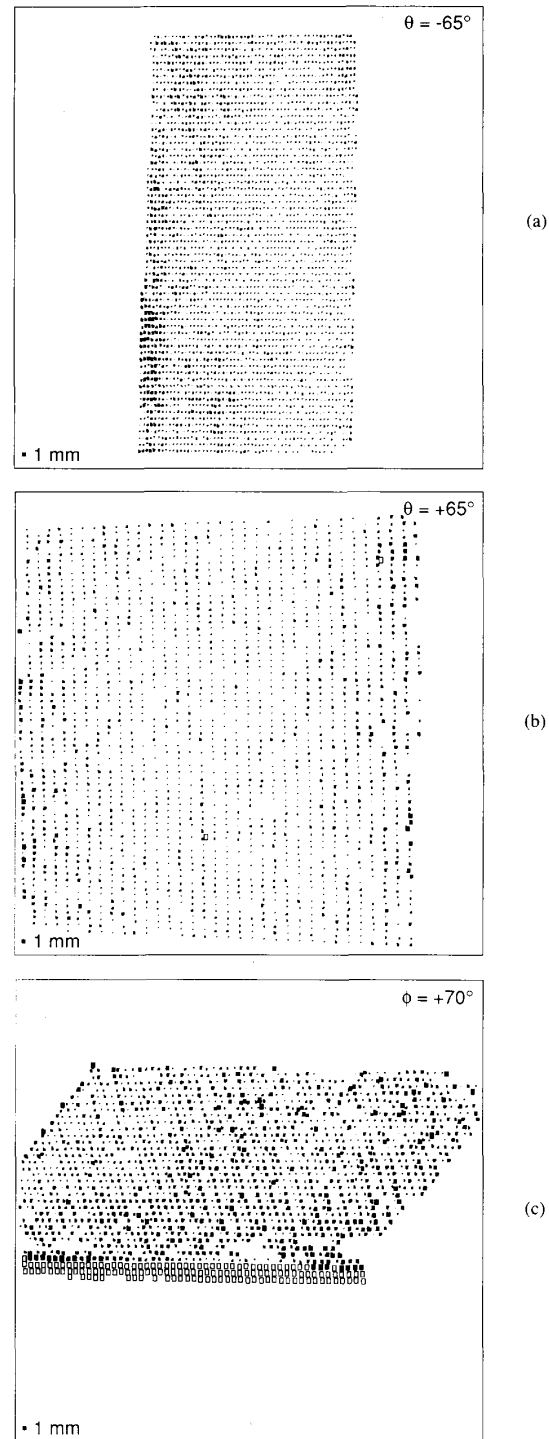[5]These connections actually correspond to the valid branches in the graph of grid points.

Fig. 14. Result images for nearly oblique target planes. Correctly identified points are denoted by a spot, the area of which is proportional to the distance to the fitted reference plane. The standard measure at the lower left corresponds to 1 mm deviation. Noncoplanar points (i.e., more than 2 mm deviation), are indicated by a box. (a), (b) Target rotated about its vertical axis by ±65°. (c) Target tilted about an axis parallel to the base line by 70°. The lower three rows of boxes correspond to points on the edge of the 18 mm thick panel.
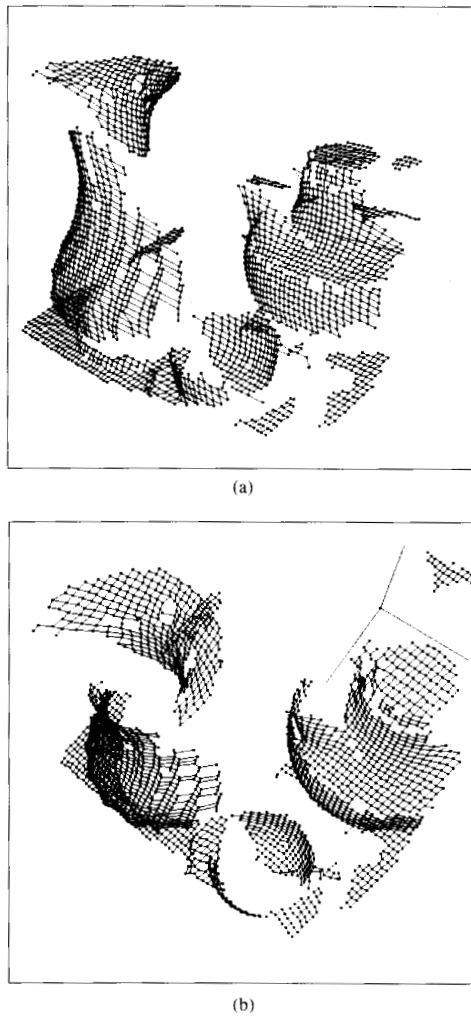
(a)

(b)

Fig. 15. Perspective views from two different view positions of the range image acquired from the vase-teapot-cup scene. The range points (indicated by square dots) have been connected to enhance visualization.

V. CONCLUSIONS

This novel approach to illumination encoding makes it possible to acquire a range image from a single camera picture. In this respect the method differs from other triangulation techniques, which take their input from multiple picture frames. The basic "snapshot" approach makes the method suited for applications in robotic assembly, where several parts of the scene might be moving, or in medical diagnostics, where it is difficult to fix the human body. The smearing effect due to the frame integration time can be entirely avoided by using high intensity pulsed illumination.

Most of the difficulties caused by the initial restriction of using only a single (monochrome) picture have been solved successfully, and fair results have been obtained with an experimental measurement system. A few topics need further investigation however. It is clear that real-
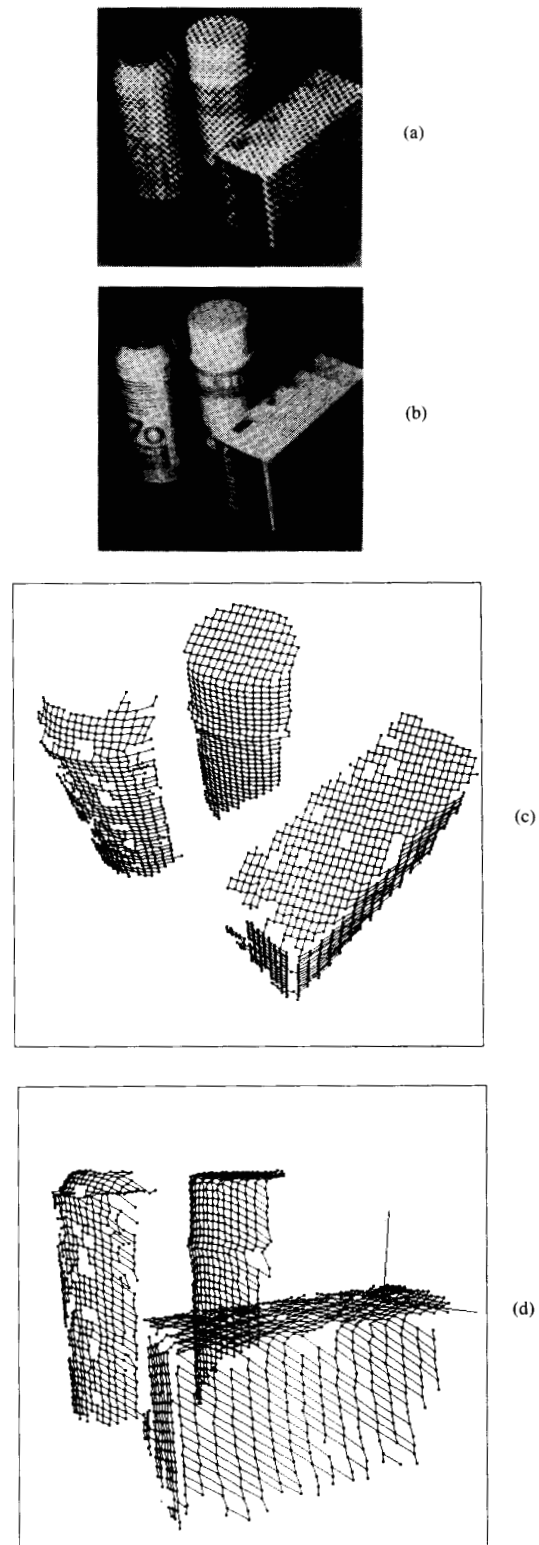

(a)

(b)

(c)

(d)

Fig. 16. Scene consisting of two spray cans and a box. (a) Illumination-encoded image. (b) Picture illustrating the scene with normal lighting. (c), (d) Perspective views of the range image.
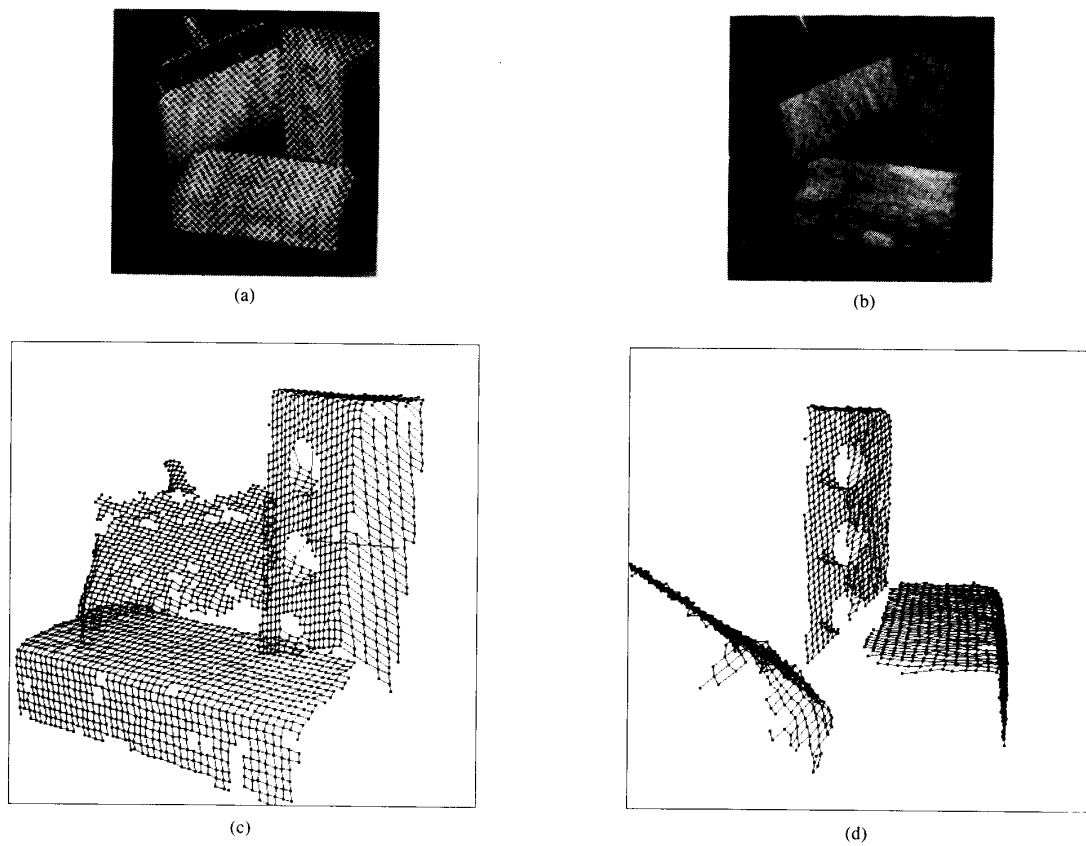
(a)

(b)

(c)

(d)

Fig. 17. A long-haired brush and two bricks. (a) Illumination-encoded image. (b) Picture illustrating the scene with normal lighting. (c), (d) Perspective views of the range image.
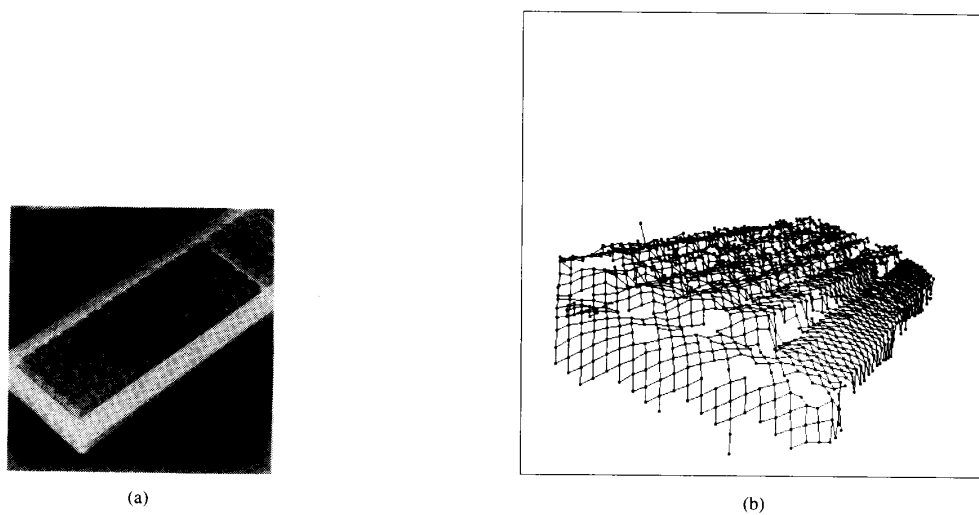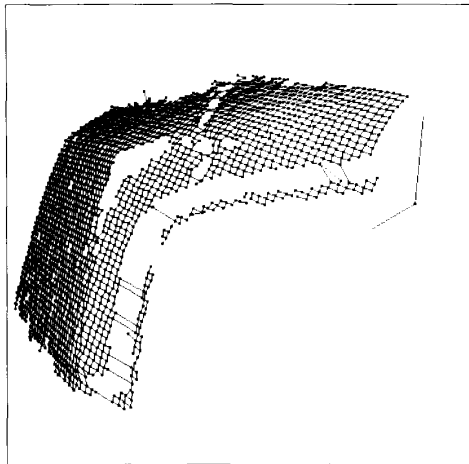

(a)

(b)

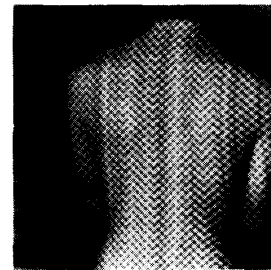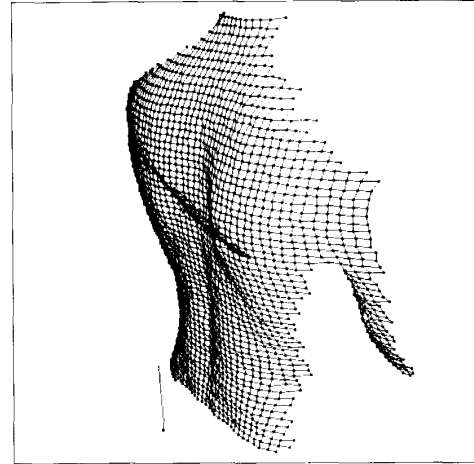Fig. 18. A keyboard. (a) Picture illustrating the scene with normal lighting. (b) Perspective view of the acquired range image.

Fig. 19. A coachwork part. (a) Picture illustrating the scene with normal lighting. (b) Perspective view of the acquired range image.

Fig. 20. Human back. (a) Illumination-encoded image. (b) Perspective view of the acquired range image.

time operation can only be achieved if computation is speeded up by a factor of at least 300. In that respect the low-level image processing routines have been conceived to be easily implementable in hardware as a pipeline of dedicated processors, involving only basic arithmetic and logical operations.

In the current prototype the range image size is limited to 64 × 63 points, which may be insufficient for some applications. One grid point corresponds to an average of 8 × 8 pixels, the input picture being 512 × 512 pixels. By reducing this ratio one could improve the lateral resolution of the range image; a few experiments that have been carried out in that perspective are reported in [32]. Performance may further be enhanced by better conditioning of the video signal, and more efficient use of the sensor surface (only 512 × 390 sensor elements are effectively used in the current prototype). The rather high susceptibility to texture interference is perhaps the most typical limitation of this nonsequential encoding technique. This problem could likely be alleviated by adding extra information channels (using a colored illumination pattern and a color camera).

REFERENCES

[1] A. Kiessling, "A fast scanning method for three-dimensional scenes," in *Proc. 3rd Int. Conf. Pattern Recognition*, 1976, pp. 586-589.
[2] P. R. Haugen, R. E. Keil, and C. Bocchi, "A 3-D active vision system," *Proc. Soc. Photo-Opt. Instrum. Engineers*, vol. 521, pp. 258-263, 1984.
[3] G. L. Oomen and W. J. P. A. Verbeek, "A real-time optical profile sensor for robot arc welding," *Proc. Soc. of Photo-Opt. Instrum. Engineers*, vol. 449, pp. 62-71, 1983.
[4] G. J. Agin and T. O. Binford, "Computer description of curved objects," *IEEE Trans. Comput.*, vol. C-25, pp. 439-449, Apr. 1976.
[5] C. G. Morgan, J. S. E. Bromley, P. G. Davey, and A. R. Vidler, "Visual guidance techniques for robot arc-welding," *Proc. Soc. Photo-Opt. Instrum. Engineers*, vol. 449, pp. 390-399, 1983.
[6] Y. Sato, H. Kitagawa, and H. Fujita, "Shape measurement of curved objects using multiple slit ray projections," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-4, pp. 641-646, Nov. 1982.
[7] K. G. Harding and K. Goodson, "Hybrid, high accuracy structured light profiler," *Proc. Soc. Photo-Opt. Instrum. Engineers*, vol. 728, pp. 132-145, 1986.
[8] G. Bickel, G. Häusler, and M. Maul, "Triangulation with expanded range of depth," *Opt. Eng.*, vol. 24, pp. 975-977, Nov. 1985.
[9] M. Rioux, "Laser range finder based on synchronized scanners," *Appl. Opt.*, vol. 23, pp. 3837-3844, Nov. 1984.
[10] M. Rioux and F. Blais, "Compact three-dimensional camera for robotic applications," *J. Opt. Soc. Amer. A*, vol. 3, pp. 1518-1521, Sept. 1986.
[11] M. Idesawa, "A new type of miniaturized optical range sensing

scheme," in *Proc. 7th Int. Conf. Pattern Recognition*, Aug. 1984, pp. 902-904.

[12] F. Röcker and A. Kiessling, "Methods for analyzing three-dimensional scenes," in *Proc. 4th Int. Joint Conf. Artificial Intelligence*, 1975, pp. 669-671.

[13] S. T. Barnard and M. A. Fischler, "Computational stereo," *Comput. Surveys*, vol. 14, pp. 553-572, Dec. 1982.

[14] J. B. K. Tio, C. A. McPherson, and E. L. Hall, "Curved surface measurement for robot vision," in *Proc. IEEE Conf. Pattern Recognition and Image Processing*, June 1982, pp. 370-378.

[15] G. Stockman and G. Hu, "Sensing 3-D surface patches using a projected grid," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 1986, pp. 602-607.

[16] J. Labuz, "Camera and projector motion for range mapping," *Proc. Soc. Photo-Opt. Instrum. Engineers*, vol. 728, pp. 227-234, 1986.

[17] B. Carrihill and R. Hummel, "Experiments with the intensity ratio depth sensor," *Comput. Vision, Graphics, Image Processing*, vol. 32, pp. 337-358, 1985.

[18] J. Tajima, "Rainbow range finder principle for range data acquisition," in *Proc. IEEE Workshop Industrial Applications of Machine Vision and Machine Intelligence*, Feb. 1987, pp. 381-386.

[19] M. D. Altschuler, B. R. Altschuler, and J. Taboada, "Measuring surfaces space-coded by a laser-projected dot matrix," *Proc. Soc. Photo-Opt. Instrum. Engineers*, vol. 182, pp. 187-191, 1979.

[20] M. Minou, T. Kanade, and T. Sakai, "A method of time-coded parallel planes of light for depth measurement," *Trans. IECE Japan*, vol. E64, pp. 521-528, Aug. 1981.

[21] S. Inokuchi, K. Sato, and F. Matsuda, "Range-imaging system for 3-D object recognition," in *Proc. 7th Int. Conf. Pattern Recognition*, 1984, pp. 806-808.

[22] K. Sato, H. Yamamoto, and S. Inokuchi, "Tuned range finder for high precision 3D data," in *Proc. 8th Int. Conf. Pattern Recognition*, Oct. 186, pp. 1168-1171.

[23] K. L. Boyer and A. C. Kak, "Color-encoded structured light for rapid active ranging," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-9, pp. 14-28, Jan. 1987.

[24] W. W. Peterson and E. J. Weldon, Jr., *Error-Correcting Codes*. Cambridge, MA: M.I.T. Press, 1972, p. 124.

[25] S. W. Golomb, *Shift Register Sequences*. San Francisco: CA: Holden-Day, 1967.

[26] F. J. MacWilliams and N. J. A. Sloane, "Psuedo-random sequences and arrays," *Proc. IEEE*, vol. 64, pp. 1715-1729, Dec. 1976.

[27] ——, *The Theory of Error-Correcting Codes*. New York: North-Holland, 1978, ch. 14.

[28] E. R. Berlekamp, *Algebraic Coding Theory*. New York: McGraw-Hill, 1968.

[29] P. Vuylsteke, "Een meetsysteem voor de acquisitie van ruimtelijke beelden, gebaseerd op niet-sequentiële belichtingscodering," Ph.D. dissertation, Katholieke Universiteit Leuven, Dec. 1987.

[30] R. C. Bolles, J. H. Kremers, and R. A. Cain, "A simple sensor to gather three-dimensional data," S.R.I. Tech. Note 249, 1981.

[31] D. F. Rogers and J. A. Adams, *Mathematical Elements for Computer Graphics*. New York: McGraw-Hill, 1976, pp. 81-83.

[32] P. Van Ranst, "Drie-dimensionale beeldacquisitie met behulp van gestructureerde belichting," M.Sc. thesis, Katholieke Universiteit Leuven, 1987.

P. Vuylsteke received the Candidate, M.S.E.E., and Ph.D. degrees from the Catholic University of Leuven, Belgium, in 1977, 1980, and 1987, respectively.

From 1980 until 1988 he was with the E.S.A.T. Laboratory, Faculty of Engineering, at the Catholic University of Leuven. While there, he participated in realizing an image computer system, and developed a new technique for 3-D image acquisition. In May 1988 he joined the R&D division of Agfa-Gevaert N.V., Mortsel, Belgium. His research interests include computer vision and image analysis.

A. Oosterlinck (S'73-M'80) received the Ph.D. degree from the Catholic University of Leuven, Belgium, in 1977, after having worked with several laboratories, including the Jet Propulsion Lab., PA. He is now a Professor of Electrical Engineering at the Catholic University of Leuven, where he is director of the E.S.A.T. Laboratory of Electronics, Systems, Automation, and Technology. He is also the founder of an image processing company for automatic visual inspection. His work has been published in more than one hundred international publications. His special research interests include image processing, picture coding, medical imaging, robot vision, and visual inspection, and the development of vision hardware.