Generation and Use of Color Pseudo Random Sequences for Coding Structured Light in Active Ranging

Heinz Hügli, Gilbert Maître

University of Neuchâtel, Institut de microtechnique
Rue Breguet 2 CH-2000 Neuchâtel, Switzerland

ABSTRACT

When using structured light for the measurement of three dimensional objects, the observed stripes of light must be indexed, that is, the correspondance between projected light stripes and the observed stripes must be established. Basically, time, spatial and color coding can be used to characterize the light stripes. In this study, we use color coding and generate the projected light stripes as a sequence of colors showing subsequences of length N which are all different and thus can be indexed by analysis of the observed stripe sequence. The sequence generation used follows the methods known for generating pseudo random sequences. We present also several indexing rules and analyse their behaviour in a measurement setup exhibiting stripe deletions and permutations. A parametric method is proposed which can accomodate scenes with different object coarseness.

1 INTRODUCTION

Active ranging methods for measuring three-dimensional (3D) objects have been widely studied in the context of computer vision applications [4]. Among direct active ranging methods, a straightforward method is certainly the one using a video camera and planes of light intersecting the object, drawing stripes on its surface. The location of the light planes being known, one can calculate the range of illuminated points by simple triangulation. However, this assumes that the identification of the different stripes is possible. Most methods used for stripe identification are based on time multiplexing or coding. In the simplest one, a single stripe is projected at once and stripe identification is trivial. This was done by Shirai [2], Agin and Binford [3] among others. An alternative method, which reduces the number of images, uses binary coding to encode the stripes [5] : the stripes are projected several times and several stripes at a time in such a way that the resulting time sequences of binary values (illuminated or not) uniquely identify each stripe. However, this technique is not satisfactory if one wishes to measure the scene in a single image, as it may be necessary for fast recording, in applications involving moving objects for instance.

Recently, such a technique has been suggested by Boyer and Kak [1]. Using color encoding, they are able to range a scene with a single image. The principle followed by these authors is to use a block of color stripes to encode the index. The code is bound to the block and a kind of synchronisation is required for decoding. The present paper proposes to use a position independant coding scheme as it is known from the theory on random sequences [7]. In the following, we analyse color encoding by use of random sequences. The result is a new way of coding which is satisfactory both from a theoretical and a practical point of view.

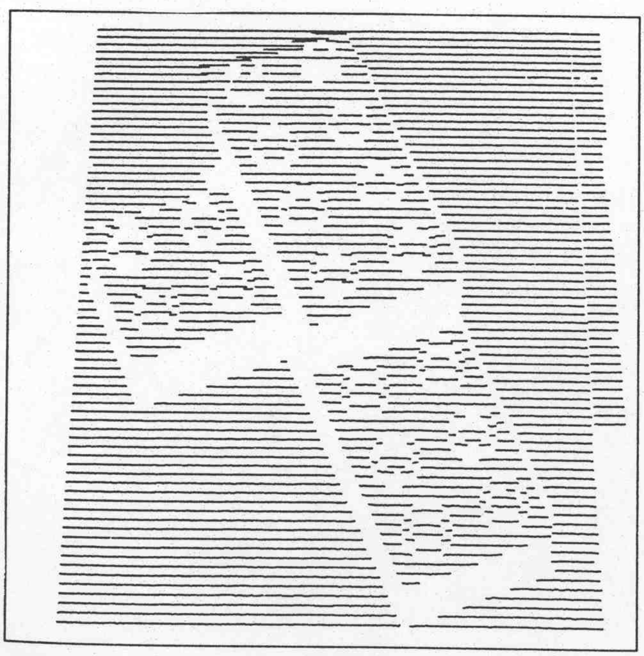

Figure 1. Stripes on a 3D object

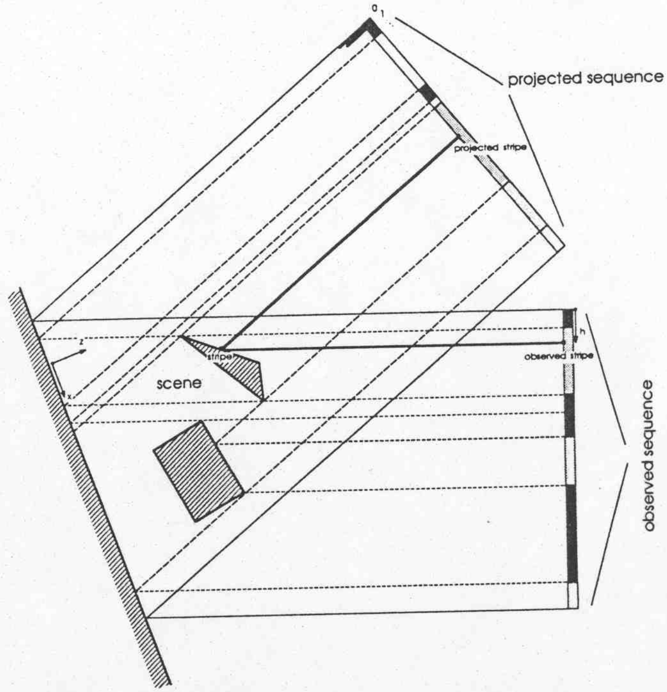

Figure 2. Model for the 3D measurement setup

2 RANGE FINDING AND STRIPE INDEXING

Direct ranging with structured light is based on triangulation. The basic setup includes a stripe projector and a 2D-camera. The projector projects several stripes of light at various positions referenced by k=1, 2, 3, ... The 3D coordinates of the illuminated points on the object surface are computed by intersecting the plane of light with the line leaving the camera at the measured screen coordinates (i,j). Now, if several planes of light are available, they appear as several stripes simultanously in the image (fig. 1). Indetermination occurs and must be removed by assigning, to each observed stripe, an index k which identifies the plane of light to which it belongs. Finding the light source index for each observed stripe is known as the *indexing problem.*
Indexing the stripes is usually not a trivial problem, because the order in which the stripes are observed is not necessarily the same as the order in which they are projected. This is because certain stripes are missing, due to occlusion and some other are permuted in the sequence, due to parallax.

In summary, ranging a point when several stripes are projected at once starts by identifying or *indexing* each stripe by the index of the plane of light it belongs to.

3 STRIPE INDEXING BY SPATIAL COLOR-CODING

Using color seems to be a straightforward method to perform stripe identification. A different color could be given to each stripe which would characterize it uniquely. Unfortunately, it is not applicable in practice because projected colors are altered or even completely absorbed when reflected by objects and are therefore often confused or missed.
Nevertheless colors can be recognized correctly in not too extreme cases if the colors are sufficiently different from each other. This reduces the number of colors to a few ones and also drastically limits the number of stripes to a few ones.
Spatial color coding is a solution to this problem. The principle is to use several neighbouring stripes for building a color sequence which encodes the stripe index and uniquely identifies the stripes. The advantage is that the number of indexes is now much larger than the number of colors. The price to pay however is a slightly reduced scene resolution, because now more than one stripe is required for indexing. Practically, this limitation means that for a surface element of the object to be indexed correctly, several stripes must cover it, which requires an object surface sufficently large compared to the structure of the projected pattern.

4 CODING BY N-UNIQUE COLOR SEQUENCES

Hereafter, we show spatial color-coding using the principle of N-unique color sequences. The problem is first formalized and analyzed, then practical indexing rules are given.

4.1 Definitions

4.1.1 Sequences and subsequences. Given elements x(m), we note a sequence of M elements

$$\mathbf{X}_M = x(1), x(2), x(3), \dots, x(M) = (x(m) \mid m=1\dots M)$$

and a subsequence of it, given by L elements centered around element x(m):

$$\mathbf{x}_L(m) = x(m-k_1), \dots, x(m), \dots, x(m+k_2) = (x(m+k) \mid k=-k_1\dots k_2) \quad ; \; k_1+k_2+1 = L \; ; \; m \in \mu = [\,(1+k_1)\dots(M-k_2)]$$

Centering is slightly different for subsequences of odd and even lengths. For both cases it writes:

$$k_1=\text{entier}\left(\frac{L-1}{2}\right) \; ; \; k_2=\text{entier}\left(\frac{L}{2}\right)$$

4.1.2 Color sequence is defined as a sequence $\mathbf{X}_M$ whose elements x(m) take values in the set of Q colors as

$$x(m) \in \Gamma = \{ C_1, C_2, C_3, \dots, C_Q \}$$

4.1.3 L-unique color sequence. A $\mathbf{X}_M$ color sequence is said L-unique if all its $\mathbf{x}_L(m)$ subsequences are different:

$$\mathbf{x}_L(m) \neq \mathbf{x}_L(n) \qquad ; m,n \in 1\dots M; \; m \neq n$$

Now, we show that a L-unique color sequence is also (L+1)-unique. Given $\mathbf{X}_M$, a L-unique color sequence. Comparing any two subsequences of length (L+1), we have:

$$\mathbf{x}_{L+1}(m) = \mathbf{x}_L(m), x\left(m+\frac{L+1}{2}\right) \quad \text{and} \quad \mathbf{x}_{L+1}(n) = \mathbf{x}_L(n), x\left(n+\frac{L+1}{2}\right) \quad \text{for } m \neq n \text{ and L odd}$$

Now, because of the L-uniqueness of $\mathbf{X}_M$, $\mathbf{x}_L(m)$ and $\mathbf{x}_L(n)$ are different, and so are the longer sequences $\mathbf{x}_{L+1}(m)$ and $\mathbf{x}_{L+1}(n)$, qed. The demonstration for L even is similar.

4.1.4 N-unique color sequence Given a L-unique color sequence. We can wonder if it is also (L-1)-unique. We write N, the smallest value of L for which a given sequence is L-unique and not (L-1) unique.

4.2 Position function of a N-unique color sequence

As, by definiton, all N-subsequences of a N-unique sequence are different, it is possible to find the position in the sequence of any of its N-subsequences by observing it, provided the sequence is known a priori. Similarly, and because $\mathbf{X}_M$ is also L-unique , a position function can be defined for sequences of length L>N.

4.2.1 Definition To do so, we define a position function on $\mathbf{X}_L$, L≥N and give it the value m, index of the equal subsequence $\mathbf{x}_L(m)$ found in the original N-unique sequence $\mathbf{X}_M$:

$$P : \Gamma^L \rightarrow [\mu \cup \{nil\}]$$

$$P(\mathbf{X}_L) = \begin{cases} m & \text{if } \mathbf{X}_L = \mathbf{x}_L(m);\ m = 1 \ldots M \\ nil & \text{otherwise} \end{cases}$$

4.2.2 Valid sequence The position function gives position values for all *valid* sequences $\mathbf{X}_L$ (which has an equal subsequence $\mathbf{x}_L(m)$ in $\mathbf{X}_M$) and returns nil for any *invalid* sequence (which has no equal subsequence $\mathbf{x}_N(m)$ in $\mathbf{X}_M$)

4.2.3 Color segment We call *color segment* of a given sequence $\mathbf{Y}_J$, a valid subsequence of maximal length:

$$\mathbf{y}_L(j) \mid (\, P(\mathbf{y}_L(j)) = i \neq nil\,) \text{ and } (\, x(i-k_1-1) \neq y(j-k_1-1)\,) \text{ and } (\, x(i+k_2+1) \neq y(j+k_2+1)\,)$$

n \ q	3	4	5	6	7
2	6	12	20	30	42
3	12	36	80	150	**252**
4	24	108	320	**750**	1'512
5	48	324	**1'280**	3'750	9'072
6	96	**972**	5'120		
7	192	2'916			
8	**384**	8'748			

Figure 3. Maximal length M_1 of the N-unique color sequence built with Q colors

4.3 Existence and construction of N-unique sequences

Given a set of Q colors, and N the length of the subsequences, we want to construct N-unique sequences used for projection. On one hand, combinatorial analysis tells us that there is a number of different N-subsequences equal to Q^N. On the other hand, the theory on pseudo-random sequences tells us that the sequence containing exactly once all different subsequences exists and that it can be generated [7]. Considering $\mathbf{X}_M$ as a wrap-around sequence and keeping in mind that a different N-subsequence is associated to each position, we conclude that the maximum length of a N-unique sequence is

$$M_0(Q,N) = Q^N$$

The practical case is somehow different. In order to make stripe discrimination easier and interstripe gap superfluous, we require, as in [1], neighbouring stripes of different color. It means $x(i) \neq x(i+1)$; i=1,2,3,.... This reduces the number of different N-subsequences, and consequently the maximum length of a N-unique color sequence to

$$M_1(Q,N) = Q(Q-1)^{(N-1)}$$

Another aspect is the existence of such sequences. We did not prove its existence in general but used an algorithm capable of finding sequences of maximal length for all cases of practical importance. Bold numbers in figure 3 indicate the upper limits of the range for which the algorithm was tested and showed successful. We did not experience any fail.

5 INDEXING WITH POSITION FUNCTION

5.1 Principle

The basic idea is to use, as structured light, a sequence of color stripes which is a known N-unique color sequence. The position or index of an observed short subsequence can thus be found by simple use of the position function.

In the remainder, we use $\mathbf{T}_I$ and $\mathbf{R}_J$ to describe the transmitted and received sequences of stripes whereas the sequences $\mathbf{X}_I$ and $\mathbf{Y}_J$ will be used to describe their respective colors.

An indexing rule is one which, from the observation of a received and possibly disturbed color sequence $\mathbf{Y}_J$, gives estimates $\hat{\imath}(j)$; $j = 1 \ldots J$ for the indexes of the observed stripe r(j). The index refers to the transmitted stripe $t(\hat{\imath}(j))$. Ideally, it points to the transmitted stripe it belongs to.

5.2 T to R mapping

Formally, we note the sequence of the I transmitted stripes $\mathbf{T}_I$ and the sequence of received stripes $\mathbf{R}_J$:

$$\mathbf{T}_I = t(1), t(2), \ldots, t(I) \qquad \text{and} \qquad \mathbf{R}_J = r(1), r(2), \ldots, r(J)$$

If t(i) transforms in r(j) we call them corresponding stripes and write: t(i) -> r(j).

In the ideal case, each transmitted stripe transforms in the received stripe of same index: t(i) -> r(i) ; i=1…I ,and indexing can be performed simply by direct application of the position function.

In practical cases, the received sequence will be different from the transmitted one. The geometry of the measurement setup as well as the objects to be measured will in general highly influence the number and order of received stripes as shown in figure 2. There is both deletion and permutation of transmitted stripes. Some stripes are deleted because of occlusions.

Some other stripes or subsequences of stripes may be permuted due to parallax which results from the fact that both illumination and observation present different incidence angles.

5.2.1 Index function This scene-dependent sequence transform is a mapping of **T** to **R** such that:
1) Every received stripe r(j) of $\mathbf{R}_J$ is the mapping of some unique stripe t(i) of $\mathbf{T}_I$ described by the index(j) function:

$$t(\text{index}(j)) \to r(j) \qquad j=1, 2, 3, \ldots, J$$

In other words, each r(j) has exactly one corresponding t(i) whereas each t(i) can have zero, one or more (i.e. in the case of a stripe split by two surfaces) corresponding r(j).
2) Corresponding stripes have the same color

Note that this model does not take care of errors such as incorrectly detected colors.

5.2.2 Undisturbed stripe sequence and stripe segment Another possible description of the mapping is in terms of its undisturbed stripe sequences as follows. We define as *undisturbed*, transmitted and received stripe subsequences which have stripes mapping onto each other:

$$\mathbf{t}_L(i) \to \mathbf{r}_L(j) \iff (t(i+k) \dashrightarrow r(j+k) \mid k= -k_1 \ldots k_2)$$

We call *stripe segments*, undisturbed stripe subsequences of maximal length:

$$\mathbf{t}_L(i) \to \mathbf{r}_L(j) \iff (t(i+k) \dashrightarrow r(j+k) \mid k= -k_1 \ldots k_2) \text{ and } (t(i-k_1-1) \not\to r(j-k_1-1)) \text{ and } (t(i+k_2+1) \not\to r(j+k_2+1))$$

Using this definition of a stripe segment, a **T** to **R** mapping can also be described by the sequence of its received stripe segments

$$\mathbf{R}_J = \mathbf{r}_{L1}(j_1),\ \mathbf{r}_{L2}(j_2), \ldots$$

With the perspective of using the position function to index subsequences and stripes, following proprieties apply:
- application of the position function to the color sequence of an undisturbed received subsequence always returns the position of the corresponding subsequence, provided $L \geq N$

- application of the position function to the color sequence of a disturbed received subsequence returns a position value which may be nil, but may also indicate a position if the color subsequence is valid

Consequently, the set of solutions $\{P(\mathbf{x}_L(j)$; L=N,N+1,N+2, ...} is a superset of the solutions for all undisturbed stripe subsequences and *a fortiori* for all stripe segments. Selecting the good solutions is the subject of the following sections.

5.3 Indexing subsequences

Given the received color sequence $\mathbf{Y}_J$ and the position function $P(\mathbf{X}_L)$; L=N, N+1, N+2, ... of the transmitted N-unique color sequence $\mathbf{X}_I$, we apply the following rules for indexing the subsequences.

5.3.1 Rule R1 Application of ideal indexing as if all received subsequences were undisturbed. To do so we observe the subsequences $\mathbf{y}_N(j)$ and apply the position function:

$$p_N(j) = P(\ \mathbf{y}_N(j)\ )$$

Undisturbed N-subsequences are indexed correctly. Disturbed subsequences are rejected only as far as non-valid color subsequences are generated, resulting in a nil value for $p_N(j)$. Rejection may be weak, because there are usually no or only few nil values in the position function $P(\mathbf{X}_N)$. Rule 2 is a possible way to improve rejection.

5.3.2 Rule R2 It uses the fact that the $P(\mathbf{X}_L)$ functions have an increased number of nil values when L increases, and therefore reject disturbed subsequences much better. The following rule applies:

$$p_L(j) = P(\ \mathbf{y}_L(j)\ )\ \ ;\ L>N\ ;\ j=1...J$$

Undisturbed L_x-subsequences are indexed correctly as far as $L_x \geq L$. Disturbed subsequences are rejected only as far as non-valid color subsequences are generated, resulting in a nil value for $p_L(j)$. Rejection is improved as far as the number of nil values in the position function $P(\mathbf{X}_L)$ is increased.

5.4 Indexing stripes

The position function $P(\mathbf{y}_N(j))$ gives an estimated index which applies somehow to all stripes of the N-subsequence. Inversely, a stripe gets N indexes $i_{j-k}(j) = p_N(j-k) + k$ which we note

$$\hat{\imath}(j,k) = p_N(j-k) + k\ \ ;\ k= -k_1...k_2$$

In one approach, all different indexes are given as possible solutions for the stripe. In another approach, developped in the next section, we present and analyse different indexing rules which decide for a single index.

5.5 Stripe indexing rules

5.5.1 Searching color segments We are looking for color segments in the received sequence. The example in figure 4 represents the color segments computed for a given position sequence ($p_N(j)$ | j=1...J). Each non-nil $p_L(j)$ value is represented by a circle at level L. It is the top of the pyramide built on the color sequence of length L which supports it. Formally, the envelope curve of the small circles is

$$l(j) = L \mid (p_L(j) \neq \text{nil})\ \text{and}\ \ (p_{L+1}(j) = \text{nil})$$

and its maxima define the top of the pyramides for color segments. Thus we know the color segments and their positions. The problem remains to assign each stripe to a given segment.

In order to do so we characterize a segment by its length L_m and one of its stripes by L_r, the relative support it gets from the segment, defined as the number of elements of the segment which support it. It is obvious that the support of a given stripe relative to a segment cannot exceed the segment length:

$$L_r \leq L_m \text{ of relative segment}$$

Now, indexing is performed based on the following facts:
- for each stripe j, there are N index candidates $\hat{\imath}(j,k)$; k=-k_1...k_2
- to each $\hat{\imath}(j,k)$ we associate the length L_m and the relative support L_r of the corresponding segment and note them $L_m(j,k)$ and $L_r(j,k)$

- as the probability for segments to be generated by noise is lower for large segments than for short ones, larger segments are given the highest confidence

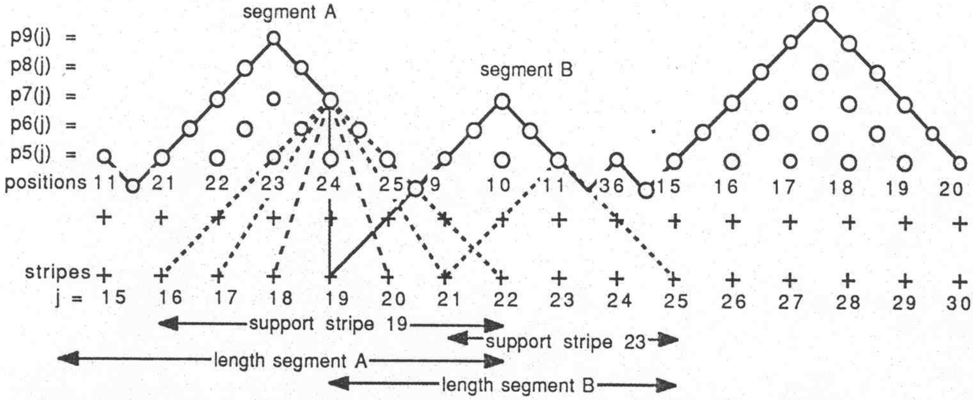

Figure 4. Example of indexing a color sequence defined by the positions $(p_5(j)| \ j=15\ldots30)$

<u>5.5.4 Rule S2</u> This rule accepts, among the index candidates $\hat{\imath}(j,k)$, the one index with
1) a sufficiently large attributed segment length and a sufficiently large attributed relative support:

$$L_m(j,k) \geq L_{m0} \qquad \text{and} \qquad L_r(j,k) \geq L_{r0} \geq N \ ; L_{r0} = 1,3,5,7\ldots$$

The requirement $L_{r0} \geq N$ garanties the uniqueness of the solution with this rule. Using an equivalent form, rule S1 writes:

$$\hat{\imath}(j) = \begin{cases} p_{L_{r0}}(j) & \text{if } L_m(j,k) \geq L_{m0} \\ \text{nil} & \text{otherwise} \end{cases} \qquad \text{with} \quad L_{r0} \geq N \ ; L_{r0} = 1,3,5,7\ldots$$

<u>5.5.3 Rule S2</u> This rule accepts index candidates $\hat{\imath}(j,k)$ in a way similar to rule S1. Here we consider the case of the several possible solutions arising when L_{r0} can have values below N.

From $\hat{\imath}(j,k)$; $k=-k_1\ldots k_2$ we eliminate the short segments:

$$\hat{\imath}'(j,k) = \begin{cases} \hat{\imath}(j,k) & \text{if } L_m(j,k) \geq L_{m0} \\ \text{nil} & \text{otherwise} \end{cases}$$

and, among the remaining solutions, choose the one with best support

$$\hat{\imath}(j) = \hat{\imath}(j,k^*) \quad ; k^* = \text{argmax}_k[\ L_r(j,k)\ ]$$

No decision is taken in the case of segments of equal support. The estimated index is nil.

<u>5.5.4 Rule S3</u> This rule selects an index among index candidates $\hat{\imath}(j,k)$ in a way similar to rule S2. The difference with S2 is that rule S3 adds a preprocessing which eliminates relative short segments. We designate a segment of length L_m as relative short, if all the stripes supporting it belong to segments with length > L_m.

<u>6 INDEXING TESTS</u>

<u>6.1 Simulation</u>

A series of simulations were conducted in order to test and compare the performance of the various indexing rules.

Basically, the simulation considers $\mathbf{X}_M$ as a N-unique transmitted sequence of the type which does not admit neighbours of same color. In order to model the scene transform, we degrade the received sequence. The received stripe sequence is considered as a sequence of segments of poisson-distributed random length of mean λ. This way, the transform is described by a sole parameter λ indicating the mean segment length.

The different indexing rules are applied to a large number of received color sequences $\mathbf{Y}_J$ and the performance is measured by counting the relative number of good, bad and undecided indexes written n_{good} n_{bad}, $n_{undecided}$ according to:

$$\text{indexing} = \begin{cases} \text{good} & \text{if } \hat{\imath}(j)=\text{index}(j) \\ \text{undecided} & \text{if } \hat{\imath}(j)=\text{nil} \\ \text{bad} & \text{otherwise} \end{cases}$$

6.2 Results

The results shown are for simulations performed according the description given above with parameter values N=4 and Q= 4. The maximum sequence length is $M_1(4,4)$= 108.

i(j)	x(j)	$\hat{\imath}(j)$ for L_{m0}= 11	10	9	8	7	6	5	4
21	0	/	/	/	/	/	/	21	21
22	3	/	/	/	/	/	/	/	/
23	0	/	/	/	/	/	56	56	56
72	2	/	/	/	/	/	57	57	57
73	1	/	/	/	/	/	/	/	/
74	2	/	/	/	/	/	/	/	/
*	1	/	/	/	10	10	10	10	10
11	2	/	/	/	11	11	11	11	11
12	3	/	/	/	12	12	12	12	12
13	1	/	/	/	13	13	13	13	13
14	2	/	/	/	/	/	/	/	/
12	3	/	/	/	/	/	/	/	/
13	1	/	/	/	/	/	/	/	/
35	3	/	/	/	/	/	/	83	83
36	1	/	/	/	/	/	/	/	/
37	3	/	/	/	37	37	37	37	37
38	0	/	/	/	38	38	38	38	38
39	2	/	/	/	39	39	39	39	39
40	0	/	/	/	40	40	40	40	40
41	1	/	/	/	/	/	/	/	/
42	3	/	/	/	/	/	/	/	/
91	1	/	/	/	/	/	91	91	91
92	0	/	/	/	/	/	92	92	92
*	2	/	/	/	/	/	/	/	/
89	1	/	/	/	/	/	89	89	89
90	3	/	/	/	/	/	90	90	90
undecided		97	91	91	71	68	56	47	47
bad		1	1	1	2	3	5	11	11
good		10	16	16	35	37	47	50	50
total		108	108	108	108	108	108	108	108

Figure 4 Example of indexing with rule S1, with various L_{m0}

Figure 4 shows part of an example of indexing after rule S1 applied to segments of mean length λ= 7. Shown left are the true indexes (index(j) ; j=0...J) generated by the random process together with the colors (y(j); j=0...J).There are four colors [0...3] (Q=4) and the sequence is 4-unique. The maximum sequence length is 108. The eight columns to the right give the indexing after rule S1, with L_{m0}= 11,10,9,8,7,6,5,4 and L_{r0}=5. A nil is noted by a slash.The performance is given at the bottom and varies from (n_{good}, n_{bad}, $n_{undecided}$) = (9%,1%,89%) to (46%,10%,43%) when L_{m0} varies from 11 to 4.

6.2.1 Rule S1 et S2

The performance of rule S1 and S2 used with different parameters is shown in figure 5 and 6. The plots show the performance (n_{good}, n_{bad}, $n_{undecided}$) as a function of L_{m0}=4,...,16 for two models λ=4,7; figure 5 is for rule S1 with a support threshold L_{r0} of 7 whereas figure 6 is for rule S2 with a L_{r0} of 1.

From the results, we conclude:
- in general, we have few errors but a relative high percentage of undecided indexes
- support thresholding (L_{r0}) is good for reducing n_{bad} to a minimum
- there is a trade-off between low n_{bad} and low $n_{undecided}$ which can be adjusted by varying the theshold for segment length (L_{m0})

6.2.2 Rule S3

The performance of rule S3 used with different parameters is shown in figure 7. As above, the plots show the performance (n_{good}, n_{bad}, $n_{undecided}$) as a function of L_{m0}=4,...,16 for various models λ=4,7.

From the results, we conclude:
- the rule performs like rule S2 for large values of L_{m0}
- as expected, it performs better for small values of L_{m0}; this is because small segments which form near the transition of two long segments are rejected

7 ACKOWLEDGEMENTS

This work was supported by the Swiss National Foundation for Scientific Research under Project No. FN 2265-0.86

8 CONCLUSIONS

This paper describes the successful use of N-unique color sequences to index stripe sequences used for active ranging. Their advantage is that decoding can be applied to any subsequence, independantly of the position, provided the subsequence has a lenght $L \geq N$. The sequence generation is described. We propose also and test several indexing rules which take full advantage of the method. Rule S1 is simple to implement and can be tailored to the application by parameters. Rule S3 behaves the same when rejection is high but significantly improves the performances by removing small segments created by coding artifacts.

9 REFERENCES

1. K. L. Boyer and A. C. Kak, "Color-encoded structured light for rapid active ranging", in IEEE Trans. Pattern Anal. Machine Intell., vol. PAMI-9, pp. 14-28, January 1987.
2. Y. Shirai, "Recognition of polyhedrons with a range finder", in Pattern Recognition, vol. 4, pp. 243-250, 1972.
3. G. J. Agin and T. O. Binford, "Computer description of curved objects", in Proc. Int. Joint Conf. Artificial Intell., Stanford Univ., Aug. 1973, pp. 629-640.
4. R. A. Jarvis, "A perspective on range finding techniques for computer vision", in IEEE Trans. Pattern Anal. Machine Intell., vol. PAMI-5, pp. 122-139, March 1983.
5. J. L. Posdamer and M. D. Altschuler, "Surface measurement by space-encoded projected beam systems", in Computer Graphics and Image Processing, vol. 18, pp. 1-17, 1982.
6. D. H. Ballard and C.M. Brown, "Computer vision", Prentice-Hall, 1982
7. W.W. Peterson & E.J. Weldon Jr., "Error-correcting codes", MIT Press, Cambridge, Mass., 1972

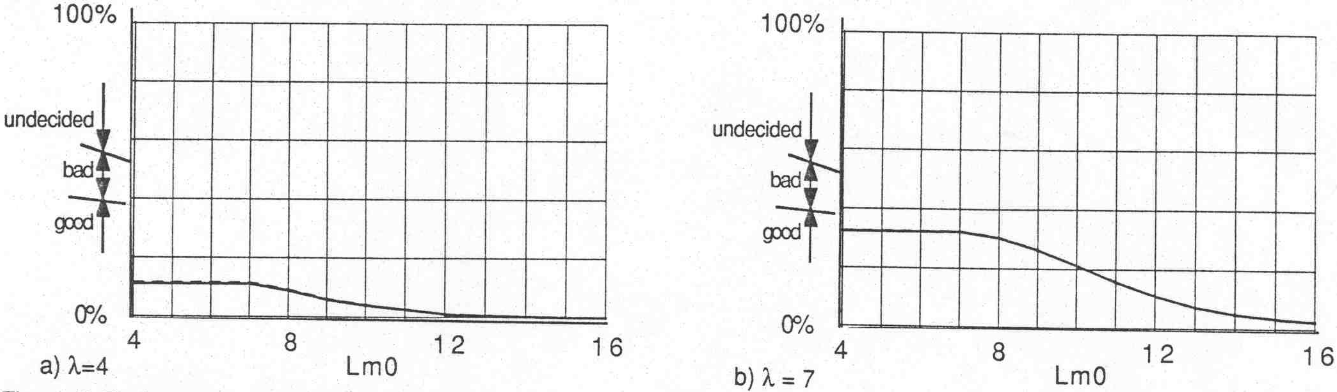

Figure 5. Performance plots for rule S1, $L_{r0}=7$;

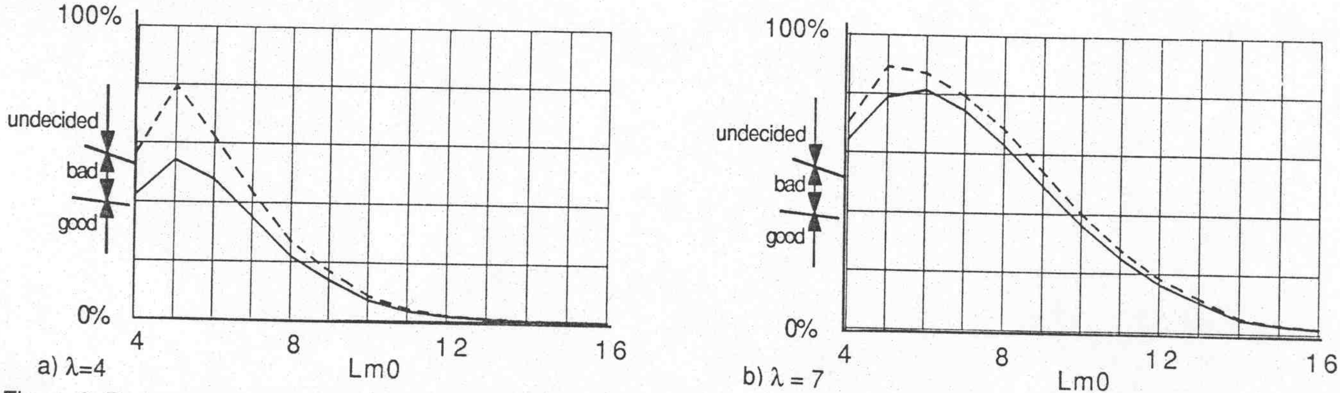

Figure 6. Performance plots for rule S2, $L_{r0}=1$;

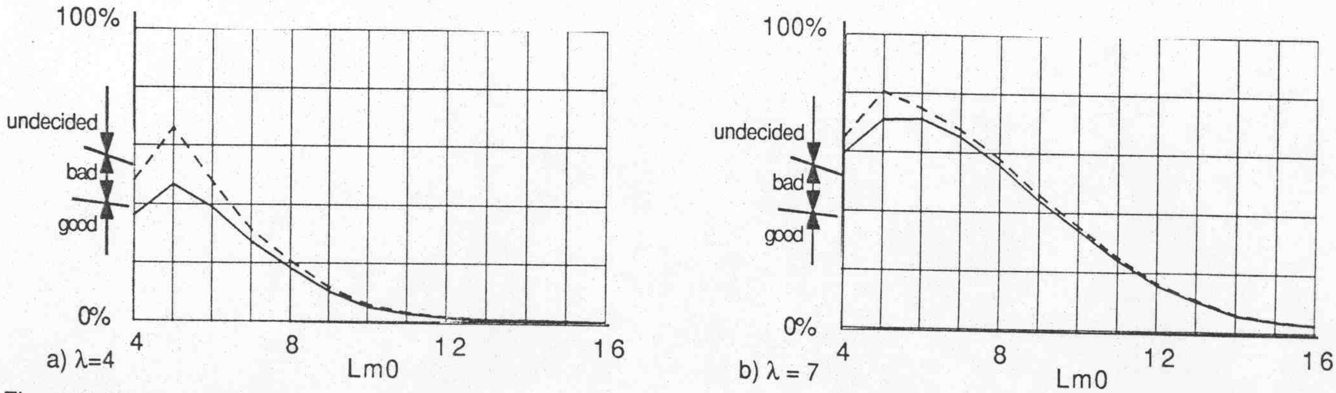

Figure 7. Performance plots for rule S3, $L_{r0}=1$;