

Now suppose that Q satisfies the condition, that is, there are two points v_i and v_j of $H(Q)$ which satisfy the inequalities. Let $v, v', w,$ and w' be points of $P(v_i, v_j), P(v_j, v_i), R(v_i, v_j),$ and $R(v_j, v_i),$ respectively, such that $v_i v v_j = \text{minint}(v_i, v_j), v_j v' v_i = \text{minint}(v_j, v_i), v_i w v_j = \text{maxext}(v_i, v_j)$ and $v_j w' v_i = \text{maxext}(v_j, v_i).$ If $v_i v v_j + v_j w' v_i < \pi,$ then the disk q whose circumference ∂q is determined by v_i, v_j and v contains Q and no point of $\bar{Q}.$ Hence, Q is a digital disk. Similarly, if $v_j v' v_i + v_i w v_j < \pi,$ then $v_i, v_j,$ and v' determine a disk q whose digital image is $Q.$ Otherwise, consider a disk q whose boundary ∂q is determined by $v_i, v_j,$ and $w.$ Every point of Q is a point of q and every point of \bar{Q} in q is on $\partial q(v_i, v_j).$ Expand q slightly by moving its center along the line bisecting $v_i v_j$ away from w while keeping v_i and v_j on ∂q and not touching any point of $R(v_j, v_i).$ Then Q is the digital image of q and so is a digital disk. This completes the proof.

Theorem 2: Given a digital region $Q,$ let v_i and v_j be any two distinct vertices of $H(Q).$ If $\text{maxext}(v_i, v_j) + \text{maxext}(v_j, v_i) \geq \pi,$ then Q is not a digital disk.

Proof: Let w and w' be points of $R(v_i, v_j)$ and $R(v_j, v_i),$ respectively, such that $v_i w v_j = \text{maxext}(v_i, v_j)$ and $v_j w' v_i = \text{maxext}(v_j, v_i).$ Denote by q the disk whose circumference is determined by $v_i, v_j,$ and $w.$ Then w' is a point of $q(v_j, v_i).$ Let p be any disk that contains v_i and v_j but not $w.$ Then $q(v_j, v_i)$ is a subset of p and so w' is a point of $p.$ Thus, there is no disk that contains Q but not any point of $\bar{Q}.$ Therefore, Q is not a digital disk.

IV. ALGORITHM

An algorithm is presented that determines whether or not a given digital region is a digital disk. The algorithm is an implementation of determining geometric properties characterized by Theorems 1 and 2.

Algorithm DIGITAL_DISK (Q):

//Given a digital region $Q,$ the algorithm determines if Q is a digital disk. If it is, the algorithm prints **True** and halts, otherwise it prints **False** and halts.//

Step 1: Construct the convex hull $H(Q).$ If $H(Q)$ has a point of \bar{Q} then print (**False**); stop.

Step 2: Construct the following two sets of digital points:

$$P = \{v_1, v_2, \dots, v_n\}, \quad \text{where } (v_1, v_2, \dots, v_n) = H(Q).$$

$$R = \{w_1, w_2, \dots, w_m\}, \quad \text{the set of boundary points of } \bar{Q}.$$

Step 3: For $i = 1$ to $n - 1$ do

for $j = i + 1$ to n do

3.1. evaluate $\text{minint}(v_i, v_j), \text{minint}(v_j, v_i),$

$\text{maxext}(v_i, v_j)$ and $\text{maxext}(v_j, v_i).$

3.2. if $\text{maxext}(v_i, v_j) + \text{maxext}(v_j, v_i) \geq \pi$

then print (**False**); stop.

3.3. if $\text{minint}(v_i, v_j) + \text{minint}(v_j, v_i) \geq \pi$

$\text{maxext}(v_i, v_j) + \text{maxext}(v_j, v_i) < \pi,$ and

$\text{minint}(v_i, v_j) > \text{maxext}(v_i, v_j)$ and

$\text{minint}(v_i, v_j) > \text{maxext}(v_j, v_i)$

then print (**True**); stop.

Step 4: Print (**False**); stop.

Suppose that the digital region Q resides in a set of $N \times N$ digital points. We assume that the region is represented by its run length code [7]. The following theorem summarizes our main result, whose proof is obvious and is omitted.

Theorem 5: Algorithm DIGITAL_DISK determines whether or not a given digital region is a digital disk requiring $O(N^3)$ time and $O(N)$ work space.

V. CONCLUSION

A definition of digital disks is given which is consistent with the definition of digital convexity and digital polygonality. With this definition, we were able to derive a few simple geometric properties that characterize digital disks. Moreover, an algorithm to determine digital disks was developed from these characterizations. The algorithm is conceptually simple and very easy to implement. Its $O(N^3)$ time complexity, however, seems rather excessive, and it is not likely that $O(N^3)$ is the lower bound.

ACKNOWLEDGMENT

The author is grateful to anonymous referees whose critical reviews of the original paper helped him to revise the paper to this improved and readable form.

REFERENCES

- [1] R. M. Haralick, "A measure for circularity of digital figures," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-4, pp. 394-396, 1974.
- [2] L. Hodes, "Discrete approximation of continuous convex blobs," *SIAM J. Appl. Math.*, vol. 19, pp. 477-485, 1970.
- [3] C. E. Kim, "On the cellular convexity of complexes," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-3, pp. 617-625, 1981.
- [4] —, "Digital convexity, straightness and convex polygons," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-4, pp. 618-626, 1982.
- [5] A. Nakamura and K. Aizawa, "On digital circles," *Comput. Vision Lab., Comput. Sci. Cen., Univ. Maryland, College Park, TR-1193, Aug. 1982.*
- [6] A. Rosenfeld, "Connectivity in digital pictures," *J. Ass. Comput. Mach.*, vol. 17, pp. 146-160, 1970.
- [7] A. Rosenfeld and A. C. Kak, *Digital Picture Processing.* New York: Academic, 1976.
- [8] J. Sklansky, "Recognition of convex blobs," *Pattern Recognition*, vol. 2, pp. 3-10, 1970.

A Two-Stage Cross Correlation Approach to Template Matching

A. GOSHTASBY, S. H. GAGE, AND J. F. BARTHOLIC

Abstract—Two-stage template matching with sum of absolute differences as the similarity measure has been developed by Vanderburg and Rosenfeld [1], [2]. This correspondence shows the development of two-stage template matching with cross correlation as the similarity measure. The threshold value of the first-stage is derived analytically and its validity is verified experimentally. Considerable speed-up over the one-stage process can be obtained by introducing only a small false dismissal probability.

Manuscript received January 23, 1983; revised May 23, 1983. This work was supported in part by NASA under Grant NGL-23-004-083.

A. Goshtasby was with the Department of Computer Science, Michigan State University, East Lansing, MI 48823. He is now with the Department of Computer Science, University of Kentucky, Lexington, KY 40506.

S. H. Gage is with the Department of Entomology, Michigan State University, East Lansing, MI 48823.

J. F. Bartholic is with the Department of Resource Development, Michigan State University, East Lansing, MI 48823.

Index Terms—Cross correlation, fast Fourier transform, Fisher's Z-transform, ranking means of independent normal populations, two-stage template matching.

I. INTRODUCTION

Template matching is the process of locating the position of a subimage inside a larger image. The subimage is called the template and the larger image is called the search area. The template matching process involves shifting the template over the search area and computing the similarity between the template and the window in the search area over which the template lies. The next step is determining the shift position where the largest similarity measure is obtainable. This is the position in the search area where the template is most likely to be located.

Major similarity measures which are used in template matching are the sum of absolute differences and the cross correlation coefficient. Sum of absolute differences as a measure is computationally fast, and algorithms are available which make the template search process even faster [3]. Assessing similarity by the correlation coefficient measure is more accurate [4], but is computationally slow.

Two-stage template matching can be used to increase the speed of the search process. In the first stage, a subtemplate is used to determine the good candidates for a match by determining those shift positions which result in a similarity measure above a threshold value. In the second stage, using those good candidates, the whole template is used to determine the best match position.

The difficulty in the two-stage template matching is in determination of an accurate correlation threshold value during the first stage. If the threshold value is too low, the search process becomes slow. The optimal match position might be missed, however, if the threshold value is too high. In this correspondence, we derive the correlation threshold value of the first stage, using subtemplate size and false dismissal probability.

II. TEMPLATE MATCHING WITH CROSS CORRELATION

The cross correlation coefficient between two windows f and g of size $N \times N$ is defined by

$$r = \frac{\sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) g(x, y)}{\left\{ \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y)^2 \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} g(x, y)^2 \right\}^{0.5}}$$

where $f(x, y)$ and $g(x, y)$ are pixel values at location (x, y) of f and g , respectively. It is assumed that f and g are normalized so that they have a mean of zero. The value of r changes between -1 and $+1$, and the closer r is to $+1$, the more similar the two windows will be.

When the search area is $M \times M$ and the template size is $N \times N$ ($N < M$), we have to compute r for every shift position (which is $(M - N + 1)^2$ shift positions):

$$r(u, v) = \frac{\sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) g(u+x, v+y)}{\left\{ \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y)^2 \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} g(u+x, v+y)^2 \right\}^{0.5}} \quad (1)$$

$$u, v = 0, 1, 2, \dots, M - N.$$

Among all the computed r 's, the one with the largest value shows the cross correlation coefficient for the best match.

We know that an operation like

$$\sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) g(u-x, v-y) \quad (2)$$

is a convolution operation and can be replaced by $FT^{-1} \{FT(f)FT^*(g)\}$, where FT is the Fourier transform operation and FT^{-1} is the Fourier inverse transform operation and f and g are square windows with dimensions of power 2. The dimensions of f and g must be equal. If they are not, they can be made equal by appending the necessary number of zero rows and zero columns to the smaller window.

If we compare the numerator of (1) with formula (2), we see that they are similar except for a change in sign. The numerator of (1) can be replaced by

$$FT^{-1} \{FT(f)FT^*(g)\} \quad (3)$$

where $*$ means the complex conjugate and takes care of the sign change in the numerator of (1), which is $+$ instead of $-$. Since Fourier transforms and Fourier inverse transforms can be computed by the fast Fourier transform (FFT) algorithm, the computation time of (1) can be reduced [5].

The computational complexity of (1) involves computation of $\sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y)^2$, which should be computed once, and computation of $\sum_{x=0}^{N-1} \sum_{y=0}^{N-1} g(u+x, v+y)^2$ which can be obtained from the squares of pixels in the search area, and which also should be computed once. By this, the computation time of the denominator turns out to be negligible with respect to that of the numerator, which involves $N^2 (M - N + 1)^2$ additions and $N^2 (M - N + 1)^2$ multiplications. The computational complexity of operation (3) is $3M^2 \log_2 M$ complex multiplications and $3M^2 \log_2 M$ real additions or $12M^2 \log_2 M$ real multiplications and $12M^2 \log_2 M$ real additions.

When M is much larger than N , the computational complexity of the direct method reduces to $N^2 M^2$ real additions and $N^2 M^2$ real multiplications, while the computational complexity of FFT algorithm involves $12M^2 \log_2 M$ real additions and $12M^2 \log_2 M$ real multiplications. We see that when M and N are both large, computation of cross correlation using the FFT algorithm is more efficient than the direct method. But, when M is much larger than N , computation of cross correlation by direct method is more efficient than the FFT algorithm.

The computation time, for a search area with $M = 32$ and template with $N = 16$ on a PDP 11/34 computer (where real addition time = $60 \mu s$ and real multiplication time = $85 \mu s$) is 10.7 for formula (1) and 6.2 for formula (3). The following sections demonstrate that by applying the two-stage template matching idea, increased speed over both the direct method and the FFT algorithm is attainable.

III. TWO-STAGE TEMPLATE MATCHING

The principle idea behind two-stage template matching is the use of a subtemplate to locate the possible positions for a match without investing time on positions which show no evidence for a match. Once this is accomplished, the objective is finding the best of the possible match positions. In [1], a subarea of the template is used as the subtemplate, while in [2], a reduced resolution template is used as the subtemplate in the first stage. We have taken the subtemplate in the first stage to be a set of n randomly selected data points from a template with N^2 data points.

Two-stage template matching using the sum of absolute differences has resulted in reduced computation time. We will show that the cross correlation coefficient, as the more accurate similarity measure, can also be used with reduced computation time, by introducing only a small false dismissal probability.

IV. THRESHOLD ESTIMATION

If the search area size and the template size are $M \times M$ and $N \times N$, respectively, the number of positions where the tem-

plate should be shifted in the search area to find the best match position is $k = (M - N + 1)^2$. Since our subtemplate is a randomly selected set of n data points from the template, by selecting a random sample for a given shift position i ($i = 1, \dots, k$), we obtain correlation coefficients r_i . It is assumed that the data points in the template and the data points in the search area over which the template lies at a given shift position i , have normal distributions and their true correlation coefficient is ρ_i (in natural life images, distribution of intensities in a window can usually be approximated by a normal distribution). Then at a given shift position i , r_i has a nearly normal distribution as long as its expected correlation coefficient ρ_i is near or equal to zero and the sample size n is large [6].

However, we wish to determine the shift position which gives the largest correlation coefficient, and the distribution of r_i when ρ_i is large is far from normal [6]. r_i can be transformed by the Fisher's Z-transform as

$$z_i = 0.5 \{ \log_e (1 + r_i) - \log_e (1 - r_i) \}. \quad (4)$$

Now whatever the value of ρ_i , z_i are nearly normally distributed with mean ζ_i and variance $1/(n-3)$ even with a sample size as small as 8 data points [6], define $\zeta_i = 0.5 \{ \log_e (1 + \rho_i) - \log_e (1 - \rho_i) \}$.

Let us assume $z_{[1]} < z_{[2]} < \dots < z_{[k]}$ denotes the order of z 's after applying the observed r 's to (4). We will also assume that $\pi_1, \pi_2, \dots, \pi_k$ are k independent and normal populations with unknown means $\zeta_1, \zeta_2, \dots, \zeta_k$ and known common variance $\sigma^2 = 1/(n-3)$, from which the x 's are observed. We wish to determine the population which has the largest mean.

In [7], [8], a subset selection procedure is given which can select a subset of the π 's to include the population with the largest mean with a minimum guaranteed probability $(1 - \alpha)$. This procedure selects all populations with observed $z_{[t]}$ such that

$$z_{[t]} \geq z_{[k]} - D\sigma \quad (5)$$

where σ^2 is the common variance of the populations, and D satisfies

$$\int_{-\infty}^{\infty} \Phi^{k-1}(z + D) \phi(z) dz = 1 - \alpha. \quad (6)$$

Φ and ϕ are, respectively, the cumulative distribution function and density function of a standard normal random variable.

The table of values for D is available for $k = 1, \dots, 10$ and $\alpha = 0.0005$ to 0.90 in [7]. For other values of k and α , D can be computed directly from (6) by numerical methods [8].

Formula (5) holds for populations that are normal and independent. Therefore, we assume that data points in each window are normally distributed and are independent of data points in other windows. Intensity distributions of windows from natural images can usually be approximated by normal distributions.

Considering the condition of independency, Fig. 1 shows an instance with positions in the search area that produce z -values equal to or greater than $z_{[k]} - D\sigma$ shaded. Windows with upper left-hand corners at points inside the shaded region may correlate with the window with $z_{[k]}$, but since these windows fall above the threshold value, it does not matter if they correlate or are independent of the window with $z_{[k]}$.

Since points outside of the shaded area are far from $z_{[k]}$, it may not be unrealistic to assume that windows with upper left hand corners at points outside of the shaded area are independent of the window with $z_{[k]}$.

To summarize, the threshold value is determined in the following manner.

- 1) Compute r_1, r_2, \dots, r_k , using the subtemplate in the first stage (there is no need to order them).
- 2) Find $r_{\max} = \max_i r_i$.
- 3) Compute $z_{[k]} = 0.5 \{ \log_e (1 + r_{\max}) - \log_e (1 - r_{\max}) \}$.

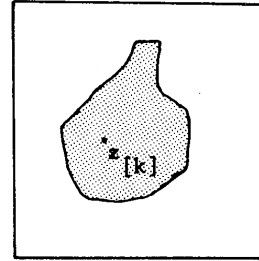


Fig. 1. The area falling above the threshold value is shaded. $z_{[k]}$ belongs to the best match window.

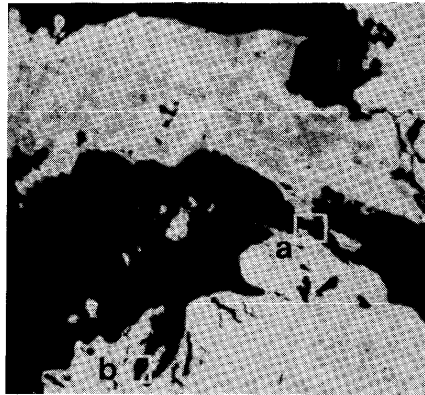


Fig. 2. Day-visible satellite image of Michigan obtained by HCMC on September 26, 1979 (scene id: A-A0518-18110-1). Windows a and b were used as search areas.

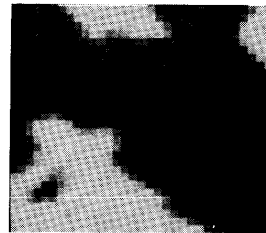


Fig. 3. Window a of image of Fig. 2.

4) Knowing $z_{[k]}$, compute $z_{[t]}$ from (5) for the given α and $\sigma = 1/\sqrt{(n-3)}$, where n is the number of data points in the subtemplate.

5) Knowing $z_{[t]}$ inversely solve (4) for $r_{[t]}$;

$$r_{[t]} = \{ \exp(2z_{[t]}) - 1 \} / \{ \exp(2z_{[t]}) + 1 \}$$

6) $r_{[t]}$ is the threshold value, and any position in the search area which has a correlation value larger than or equal to $r_{[t]}$ in the first stage should be tested in the second stage.

V. RESULTS

To measure the performance of the two-stage template matching with cross correlation and to verify the validity of the estimated threshold values, the following experiments were conducted. Heat capacity mapping mission (HCMC) satellite day-visible data was used acquired on Sept. 26, 1979 for an area over Michigan (Fig. 2). A 32×32 window (window a) was extracted from the image of Fig. 2. This is shown in Fig. 3. A 16×16 window with its upper left-hand corner

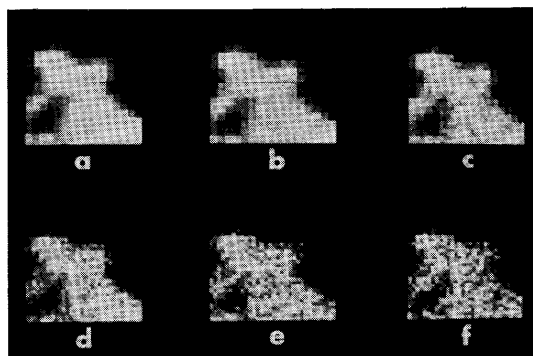


Fig. 4. (a) A 16×16 template, extracted from the search area of Fig. 3 at (11, 1). (b)-(f) show the result of adding different amplitude noise to template *a*.

TABLE I
RESULT OF TWO-STAGE TEMPLATE MATCHING WITH CROSS CORRELATION.
SEARCH AREA SIZE = 32×32 , TEMPLATE SIZE = 16×16 , SUBTEMPLATE
SIZE = 16, $\alpha = 0.05$.

Experiment #	Noise amplification factor, <i>A</i> .	Average # of positions tested in second stage in 100 tests, <i>X</i> .	# of times the best match position fell below the threshold value in first stage
1	1/5	19	0
2	2/5	34	1
3	3/5	53	0
4	4/5	57	2
5	5/5	92	3

at (11, 1) of the search area was extracted as the template [Fig. 4(a)]. The template matches completely with the window in the search area at position (11, 1).

Random noise was generated from a uniform distribution over (-1.0, 1.0) and was added to the template proportional to its intensities. If intensity at location (*x*, *y*) in the template is *I*(*x*, *y*) then noise was added to location (*x*, *y*) of the template equal to $A * I(x, y) * \text{NOISE}()$, where NOISE is the randomly generated noise over (-1.0, 1.0) and *A* is a constant amplification factor. Noise was added to a pixel proportional to its intensity value because in natural life images, noise tends to increase with the amplitude of the signal.

After adding noise to the template, with subtemplate of size 16, a search was carried out for the best match position using the two-stage template matching with cross correlation. The matching results for false dismissal probability 0.05 are shown in Table I.

Each experiment was carried out 100 times, each time with a different initial value for the noise generator. With the noise amplification factor, *A*, larger than 1.0, false matches started to appear even when the entire template was used in the search process.

The overall computation time = computation of stage 1 + stage 2. If we take a random sample of size *n* to be the subtemplate in the first stage, the computation time of the first stage = $n(M - N + 1)^2$ additions + $n(M - N + 1)^2$ multiplications. When $M = 32$, $N = 16$, and $n = 16$, the computation time for the first stage = 0.7 s.

The computation time for the second stage is a function of the number of points that fall above the threshold value, which depends on the characteristics of the template. If *X* is the number of pixels which fall above the threshold in the first stage, then the search of the second stage involves XN^2 additions and XN^2 multiplications. Using the values of *X* from Table I, the computation times of the second stage in experiments 1-5 are computed to be 0.7, 1.3, 2.0, 2.1, and 3.4 s, respectively. Adding the computation time of the first stage

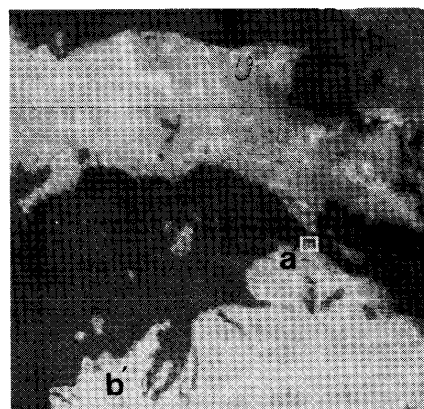


Fig. 5. Day-IR satellite image of Michigan obtained by HCMM on September 26, 1979 (scene id: A-A0518-18110-2). Windows *a'* and *b'* were used as the templates.

to the computation times of the second stage, we find the overall computation times, which are 1.4, 2.0, 2.7, 2.8, and 4.1 s for experiments 1-5, respectively. These times are about $\frac{1}{3}$ of the time needed by the FFT algorithm and about $\frac{1}{5}$ of the time needed by the direct approach.

When using this approach, we can take templates of any shape and any size. This is not true for the FFT where the template must be square and the number of pixels in each row and in each column a power of 2.

To test this approach with real data, a 16×16 window (window *a'*) was extracted from about the same area from the next day-infrared image of HCMM acquired on Sept. 16, 1979. This image is shown in Fig. 5. Using the search area *a* of Fig. 2 and template *a'* of Fig. 5, and applying the two-stage template

TABLE II
CORRELATION VALUES IN THE FIRST STAGE, FOR SEARCH AREA a SHOWN IN FIG. 2 AND TEMPLATE a' SHOWN IN FIG. 5. THE ENCLOSED POSITIONS ARE THOSE WHICH FELL ABOVE THE THRESHOLD VALUE. THESE POSITIONS ARE THE ONLY ONES WHICH SHOULD BE TESTED IN THE SECOND STAGE. THE CORRECT MATCH IS KNOWN TO BE AT (11, 1).

CORRELATION COEFFICIENTS IN THE FIRST STAGE																	
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	-0.42	-0.47	-0.60	-0.53	-0.52	-0.35	-0.43	-0.53	-0.45	-0.41	-0.33	-0.34	-0.24	-0.18	-0.16	-0.26	-0.39
1	-0.39	-0.31	-0.40	-0.53	-0.50	-0.35	-0.46	-0.38	-0.53	-0.41	-0.31	-0.40	-0.36	-0.20	-0.17	-0.32	-0.49
2	-0.19	-0.40	-0.35	-0.31	-0.39	-0.45	-0.37	-0.36	-0.45	-0.53	-0.44	-0.32	-0.34	-0.33	-0.32	-0.32	-0.37
3	-0.05	-0.13	-0.08	-0.01	-0.14	-0.26	-0.40	-0.31	-0.35	-0.42	-0.43	-0.25	-0.23	-0.34	-0.31	-0.28	-0.35
4	0.13	0.17	0.30	0.24	0.11	0.11	-0.00	0.05	0.02	-0.03	0.02	-0.01	0.03	-0.13	-0.09	-0.20	-0.25
5	0.31	0.29	0.31	0.26	0.24	0.25	0.13	0.11	0.11	0.18	0.11	0.19	0.05	0.04	-0.01	-0.01	-0.14
6	0.46	0.30	0.27	0.30	0.24	0.16	0.15	0.15	0.14	0.09	0.11	0.13	0.02	-0.02	-0.00	0.01	-0.03
7	0.47	0.40	0.44	0.30	0.16	0.16	0.14	0.23	0.12	0.09	0.07	0.12	0.03	-0.06	-0.08	0.01	-0.03
8	0.63	0.62	0.52	0.51	0.33	0.26	0.24	0.31	0.21	0.11	0.10	0.06	-0.01	0.08	-0.07	-0.07	0.05
9	0.75	0.77	0.65	0.60	0.49	0.37	0.31	0.28	0.19	0.11	0.02	0.08	0.11	0.05	-0.03	0.07	-0.02
10	0.86	0.87	0.79	0.61	0.51	0.37	0.34	0.30	0.21	0.13	0.10	0.14	0.06	0.00	0.11	0.07	-0.03
11	0.89	0.94	0.78	0.61	0.51	0.37	0.34	0.30	0.21	0.13	0.10	0.14	0.06	0.00	0.11	0.07	-0.03
12	0.78	0.87	0.65	0.61	0.58	0.59	0.48	0.47	0.34	0.20	0.11	0.09	0.10	0.16	0.15	0.06	0.14
13	0.60	0.60	0.62	0.61	0.73	0.77	0.63	0.54	0.42	0.24	0.12	0.13	0.16	0.12	0.13	0.10	0.16
14	0.38	0.58	0.63	0.68	0.73	0.73	0.70	0.55	0.47	0.38	0.25	0.13	0.11	0.13	0.11	0.18	0.13
15	0.29	0.39	0.35	0.50	0.61	0.66	0.69	0.60	0.45	0.39	0.40	0.21	0.21	0.20	0.09	0.08	0.17
16	0.12	0.30	0.26	0.32	0.48	0.53	0.53	0.55	0.58	0.50	0.32	0.28	0.31	0.33	0.23	0.09	0.10

THE THRESHOLD VALUE IS 0.5842

TABLE III
CORRELATION VALUES IN THE FIRST STAGE, FOR SEARCH AREA b SHOWN IN FIG. 2 AND TEMPLATE b' SHOWN IN FIG. 5. THE ENCLOSED POSITIONS ARE THOSE WHICH FELL ABOVE THE THRESHOLD VALUE. THESE POSITIONS ARE THE ONLY ONES WHICH SHOULD BE TESTED IN THE SECOND STAGE. THE CORRECT MATCH IS KNOWN TO BE AT (7, 3).

CORRELATION COEFFICIENTS IN THE FIRST STAGE																	
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	0.13	0.33	0.15	0.07	0.13	0.11	-0.17	-0.14	-0.19	-0.05	-0.16	-0.30	-0.17	-0.12	-0.06	0.02	0.14
1	0.06	0.13	0.17	0.12	0.17	0.13	0.04	-0.02	-0.16	-0.16	-0.27	-0.37	-0.22	-0.11	-0.07	-0.06	0.14
2	-0.05	0.02	0.08	0.09	0.30	0.35	0.10	-0.13	-0.07	-0.07	-0.24	-0.35	-0.15	-0.15	-0.11	-0.12	0.04
3	-0.07	0.00	0.13	0.11	0.34	0.47	0.18	-0.16	-0.16	-0.05	-0.04	-0.19	-0.10	-0.10	-0.12	-0.05	0.03
4	0.12	0.35	0.47	0.35	0.49	0.50	0.15	-0.12	-0.17	-0.02	-0.07	-0.18	-0.09	-0.02	-0.03	-0.02	0.07
5	0.32	0.58	0.63	0.56	0.53	0.44	0.17	-0.05	-0.16	-0.13	-0.14	-0.05	-0.01	0.10	0.11	-0.01	-0.03
6	0.46	0.65	0.79	0.81	0.69	0.52	0.29	0.11	0.03	-0.06	-0.04	0.01	0.01	0.02	0.04	-0.04	-0.02
7	0.50	0.63	0.85	0.98	0.87	0.58	0.43	0.33	0.15	-0.03	0.02	-0.02	-0.01	-0.04	-0.10	-0.06	-0.07
8	0.35	0.60	0.78	0.89	0.92	0.67	0.54	0.48	0.23	-0.05	0.08	0.01	-0.18	-0.22	-0.20	-0.12	-0.14
9	0.07	0.54	0.64	0.73	0.87	0.72	0.65	0.50	0.24	0.02	0.15	-0.09	-0.22	-0.26	-0.24	-0.23	-0.17
10	-0.10	0.22	0.55	0.66	0.70	0.66	0.60	0.39	0.18	0.04	0.01	-0.21	-0.28	-0.30	-0.28	-0.16	-0.30
11	-0.04	0.02	0.29	0.50	0.55	0.55	0.37	0.16	0.03	-0.02	-0.10	-0.25	-0.27	-0.24	-0.24	-0.24	-0.37
12	0.09	0.01	0.05	0.26	0.45	0.42	0.21	0.08	-0.05	-0.12	-0.21	-0.28	-0.23	-0.20	-0.26	-0.35	-0.26
13	0.11	0.09	0.05	0.19	0.34	0.29	0.22	0.14	-0.11	-0.33	-0.27	-0.25	-0.22	-0.27	-0.36	-0.35	-0.16
14	0.03	0.07	0.16	0.29	0.35	0.26	0.20	0.07	-0.17	-0.35	-0.28	-0.32	-0.28	-0.38	-0.32	-0.16	-0.09
15	-0.01	0.05	0.21	0.33	0.34	0.23	0.17	-0.01	-0.24	-0.25	-0.25	-0.34	-0.29	-0.25	-0.16	-0.08	-0.14
16	0.06	0.12	0.26	0.30	0.34	0.33	0.28	-0.07	-0.22	-0.20	-0.29	-0.38	-0.24	-0.14	-0.10	-0.13	-0.16

THE THRESHOLD VALUE IS 0.5087

matching with cross correlation, the results of Table II were obtained. The correct match position is known to be (11, 1). In this experiment 34 pixels fell above the threshold value, and the overall computation time for this test was 2.0 s. Using the search area b of Fig. 2 and template b' of Fig. 5, the results of Table III were obtained. Thirty-three pixels fell above the threshold value with the correct match known to be at (7, 3). Overall computation time = 2.0 s.

In the above experiments, the correct match position is assumed to be the position where the highest cross correlation value is obtained when the entire template is used in the search process.

VI. CONCLUSION

The cross correlation coefficient is a similarity measure which is widely used in template matching because of its superior accuracy over the sum of absolute differences. It is heavily used in preparation of depth maps [9] and in registration of satellite images [5]. These processes may involve hundreds of template matchings, which are quite costly. A method to speed up the template matching process with cross correlation has long been desired.

We have applied the two-stage template matching idea to the template matching with the cross correlation coefficient as the similarity measure, and have demonstrated a reduction in computation time. The correlation threshold value of the first

stage was estimated using the subtemplate size and the false dismissal probability. The validity of the process was then verified experimentally.

REFERENCES

- [1] G. J. Vanderburg and A. Rosenfeld, "Two-stage template matching," *IEEE Trans. Comput.*, vol. C-26, pp. 384-393, Apr. 1977.
- [2] A. Rosenfeld and G. J. Vanderburg, "Coarse-fine template matching," *IEEE Trans. Syst., Man, Cybern.*, pp. 104-107, Feb. 1977.
- [3] D. I. Barnea and H. F. Silverman, "A class of algorithms for fast digital image registration," *IEEE Trans. Comput.*, vol. C-21, pp. 179-186, Feb. 1972.
- [4] M. Svedlow, C. D. McGillem, and P. E. Anuta, "Experimental examination of similarity measures and preprocessing methods used for image registration," presented at the Symp. Machine Processing of Remotely Sensed Data, 1976.
- [5] P. E. Anuta, "Spatial registration of multispectral and multi-temporal digital imagery using fast Fourier transform techniques," *IEEE Trans. Geosci. Electron.*, vol. GE-8, pp. 353-368, Oct. 1970.
- [6] R. A. Fisher, *Statistical Methods for Research Workers*. New York: Hafner, 1948, pp. 175-210.
- [7] R. E. Bechhoffer, "A single-sample multiple decision procedure for ranking means of normal populations with known variances," *Ann. Math. Statist.*, vol. 25, pp. 16-39, 1954.
- [8] S. S. Gupta, "On some multiple decision (selection and ranking) rules," *Technometrics*, vol. 7, pp. 225-245, May 1965.
- [9] M. D. Levine, D. A. O'Handley, and G. M. Yagi, "Computer determination of depth maps," *Comput. Graphics Image Processing*, vol. 2, pp. 131-150, 1973.