Reprinted from

Seventh International Symposium

Machine Processing of

Remotely Sensed Data

with special emphasis on

Range, Forest and Wetlands Assessment

June 23 - 26, 1981

Proceedings

Purdue University
The Laboratory for Applications of Remote Sensing
West Lafayette, Indiana 47907 USA

A NEW CLUSTERING METHOD FOR LANDSAT IMAGES USING LOCAL MAXIMUMS OF A MULTI-DIMENSIONAL HISTOGRAM

K. MATSUMOTO, M. NAKA, H. YAMAMOTO

National Aerospace Laboratory
Tokyo, Japan

I. ABSTRACT

This paper describes a new clustering method for a huge satellite image. This method is composed of two major schemes, a multi-layered hashing scheme for multi-dimensional histogram (MDH) and a histogram clustering scheme using MDH. The MDH construction scheme works in 3 stages. In the 1st stage it constructs a few histogram layers for each part of an image, and in the 2nd and 3rd stages histogram layers are combined into one histogram. The clustering scheme searches for local maximums of MDH, and decides clusters around local maximums as sets of hyperrectangles. The major parameters of the clustering scheme are briefly discussed, and some results are also shown.

II. INTRODUCTION

Clustering technique is important as an unsupervised classifier in automated satellite image analysis. Because of the huge amount of image data, however, it is very difficult for current clustering methods to analyze more than a small sub-frame of a satellite image.

There are two major approaches to clustering. One is an ISODATA-like method that requests some reasonable initial partitions and then converges to optimal partitions using an iterative procedure. Another approach is based on observing a distribution pattern by a histogram.[3] This method segments the data in accordance with a form of the distribution pattern using some pattern recognition techniques. In the former approach, the convergence to a suboptimal segmentation is possible under reasonable initial partitions. But in this method there are many unsolved problems; for example, finding a reasonable method to estimate the initial partitions, and avoiding the possibility of multiple solutions.[1] In the latter approach, though there are some difficult problems in getting enough statistics to observe the distribution pattern and in recognizing the pattern, it has the advantage over the former approach of not needing any initial partitions.

In this paper a new method using the latter approach is described. This method derives cluster forms from an MDH. The basic idea is as follows: Since a histogram is a sample of the mixed cumulative distribution function, if we assume the probability density function of each cluster to be unimodal, the frequency value corresponding to the center of a cluster should be a local maximum. Also, the boundary between two adjacent clusters should be a valley or a ridge of the histogram. Therefore, basically, it would seem to be very natural to recognize clusters by searching for local maximums and then setting the boundaries in the valleys of the histogram.

However, there are the following problems in putting this basic idea into practice:
(1) Efficient construction scheme of an MDH.
(2) Local maximum searching algorithm using an MDH.
(3) Expression of a cluster boundary in an MDH.
(4) Processing of the sampling error of an MDH.

In section III of this paper problem (1) will be discussed, and problems (2)–(4) will be discussed in section IV.

This new method has been implemented on the large scale vector processor FACOM 230/75 APU (called the APU). Some of the results are shown in this paper.

III. CONSTRUCTION OF MULTI-DIMENSIONAL HISTOGRAM

There are two possible methods in constructing an MDH, (a) the usual method using a multi-dimensional array, and (b) a method using a histogram table. (a) is the simplest method, but some resampling process and some degradation

of data accuracy are inevitable, because the data length of an MSS pixel is 27 bits and it requires a huge memory capacity of 1.3×10^8 words even for the current LANDSAT MSS image. In method (b), the table length and the table search method may become problems. But Shlien showed that it is possible to gather more than 75%–90% pixels from a full frame LANDSAT MSS image using the double-hashing scheme and about 10,000 histogram cells.[2]

In the following discussion the MDH is constructed by the latter histogram table method using the double-hashing scheme. The reason is that it easily deals with multi-band data, does not require a huge memory, and maintains data accuracy. Even in this method, however, the following problems remain, some of which were pointed out by Shlien:

(1) Limitation of table length; especially for a full frame image.
(2) Difficulty in computing the hashing scheme for a large number of bands.
(3) Efficiency of the hashing scheme.

A. THE VECTOR FORM HASHING SCHEME FOR A LARGE NUMBER OF BANDS

On the vector processor APU, each scan line pixel is processed simultaneously. From eq. (1)–(12) hash probes' vector of each scan line pixel are calculated and shifted by eq. (12) simultaneously. Using the table address determined by the hash probes' vector, each pixel of a scan is tried sequentially, and the conflicted pixels are gathered and shifted by eq. (12) for the next trial.

$$\mathbf{V} \triangleq \{V_1, V_2, \cdots V_n\} \tag{1}$$

$$V_j = \sum_{k=1}^{m} V_j^k \cdot \alpha^{(k-1)} \qquad (j=1, \ldots, n) \tag{2}$$

$$V_j^k = \sum_{\ell=1}^{4} S^j_{\ell+4(k-1)} \cdot A^{(\ell-1)} \qquad (k=1, \ldots, m) \tag{3}$$

$$a^i \triangleq \{a_1^i, a_2^i, \cdots a_n^i\} \tag{4}$$

$$b \triangleq \{b_1, b_2, \cdots b_n\} \tag{5}$$

$$\beta_1^h = 1, \quad \beta_2^h = \mathrm{mod}(\alpha, P_h) \qquad (h=1, 2) \tag{6), (7}$$

$$\beta_k^h = \mathrm{mod}(\beta_{k-1} \cdot \beta_2^h, P_h) \tag{8}$$

$$\gamma^h_{k,j} = \mathrm{mod}(V_j^k, P_h) \tag{9}$$

$$a_j^1 = \mathrm{mod}\left\{\sum_{k=1}^{m} \gamma^1_{k,j} \cdot \beta_k^1, P_1\right\} + 1 \tag{10}$$

$$b_j = \mathrm{mod}\left\{\sum_{k=1}^{m} \gamma^2_{k,j} \cdot \beta_k^2, P_2\right\} + 1 \tag{11}$$

$$a^{i+1} = \mathrm{mod}\{a^i + b, P_1\} + 1 \tag{12}$$

where

$\alpha = 2^{28}$ $\quad A = 2^7$

P_h = the largest prime number pair which does not exceed table length.

S_ℓ^j : the spectral value of j th pixel.

n : the pixels' width of a scan line.

For a large number of bands, during the calculation of a_j^1 and b_j the m words division algorithm is necessary in the usual method, and it requires annoying calculations. However, if eq. (6)–(11) are used to calculate a_j^1 and b_j, even for a large number of bands, only a slight additional computing time is needed. Since β_k^h can be determined previously, only three (+,x,÷) additional single precision operations are needed for each additional word expressing a hash vector V. By eq. (6)–(11), the hash probes can be calculated efficiently even for a large number of bands and even with a small word-length computer.

B. MULTI-LAYERED HASHING SCHEME

The degradation of the efficiency of a hashing scheme becomes a serious problem when constructing the MDH of a huge satellite image using a limited histogram table length. The degradation of the efficiency and the difficulty of a new histogram cell's registration become more serious as the number of processed scan lines increases, and as the number of blank entries in a histogram table decreases (Fig. 1).

The usual way to maintain hashing scheme efficiency is to remove cells with less frequency from the table and to register all other cells again by the same hashing scheme, which hereafter will be called "re-hashing". But this simplest way does not work well, because during the re-hashing process some cells can not be registered again, and because once the re-hashing is activated, it is activated for almost all scan lines. Especially, re-hashing activation in almost every scan indicates the possibility of missing a relatively large cluster which appears in the latter part of an image.

To deal with this problem, the following multi-layered hashing scheme was developed.

step 1: Construct the MDH along scan lines by the usual hashing scheme using finite probes. When a certain condition is satisfied, complete a current histogram layer, and store its table in file A. Then, clear the table to begin the next histogram layer construction. While repeating this process, also store the rejected pixels in file B.

step 2: After processing all pixels of an image, get the histogram tables of each layer to add one by one sequentially by the double-hashing scheme using the same

length table. In the case of collisions, however, for the current cell take away the table entry of a cell which has the least frequency value along the collision chain. Also during this process, store the rejected or replaced cells in file B.

step 3: After adding the histograms of all layers, try again to register all pixels and cells in file B into the current histogram table by the usual double-hashing scheme.

The basic idea of this method is the combination of high speed processing of the huge, raw-image data (step 1), and the elaborate and time consuming process of selecting the larger frequency cells for the compressed data (step 2). This combination of step 1 and step 2 not only saves computing time, but also saves relatively large clusters or cells that appear in the latter part of an image.

As the possible conditions in step 1, **(a)** the number of used entries in a table, and **(b)** the number of rejected pixels, are tested. A suitable condition is to have the ability to maintain high efficiency while generating a small number of histogram layers. However, these two conditions are not compatible. A few results in Fig. 1-c,d show that **(b)** has a linear relationship to the efficiency, but **(a)** does not have, and also that **(a)** incurs the possibility of completing a layer too early, before the degradation of efficiency occurs. Therefore **(b)** is more suitable.

Theoretically, without step 3 the resulting histogram table is not a true sample of the MDH because, after completion of step 2, the possibility remains that some pixels or cells in file B have corresponding cells in the resulting table. However, the amount of such data is very small, as shown in Table 1, and cost performance of step 3 is rather low. Therefore, it is possible to take away step 3, when higher speed processing is required.

Table 1 is an example of constructing an MDH for a full frame image of a LANDSAT scene ID.1145-00542. The effect of this method is remarkable. In only 57.3% of the computing time, it gathers 25.2% more data than the basic method using the same table length.

IV. MULTI-DIMENSIONAL HISTOGRAM CLUSTERING

For an MDH and for clusters contained in a raw image data, the following are assumed:

(A1) An MDH is good enough to observe the distribution pattern of a raw image.

(A2) Each cluster has a unimodal distribution.

(A3) The frequency difference, between a peak of a cluster and valleys corresponding to the cluster's boundaries, is larger than the sampling error.

The basic idea described here is to search for a local maximum of an MDH as a peak of a cluster, to gradually extend the cluster's area from the peak, and to determine its boundaries at the points where it connects to the area of another cluster. In this idea, the recognition of a true peak and extention of an area are the most important processes. The major reason for not recognizing a true peak is the appearance of false peaks due to sampling error. The extention process is deeply related to the expression of a cluster's area.

The algorithm proposed here simultaneously carries out the three functions: searching for local maximums, determining the boundaries of clusters, and rejecting false peaks due to sampling error. Basic terminologies are defined as follows:

cluster: A *candidate* whose smallest depth of valleys surrounding it is more than **N**.

candidate: A set of adjoining but nonintersecting *areas* which is developed around an *isolated cell.*

isolated cell: A histogram cell which does not connect to any *area* of *clusters* or *candidates*.

The algorithm is as follows (Fig. 2):

step 1: Sort the MDH in descending order of frequency.

step 2: Set processing level $\mathbf{L_1}$ at the most frequent value of the MDH, and make a *candidate area* of the "record" $\mathbf{L_1}$ around the most frequent cell.

step 3: Update the current processing level, $\mathbf{L_{i+1} = L_i - \Delta L_i}$.

step 4: Establish some *candidates* as *clusters* if their "records" are older than $\mathbf{L_{i+1} + N}$.
This condition means that the smallest depth of valleys exceeds threshold **N**.

step 5: Carry out step 6 and step 7 sequentially for the cells whose frequency value **f** is $\mathbf{L_i > f \geq L_{i+1}}$.

step 6: Look over for the spatial *connectedness* between the current cell and the *areas* of *clusters* and *candidates*.

step 7:
(case: *isolated*) Make a new *candidate area* of a "record" $\mathbf{L_{i+1}}$ around the cell.
(case: *connected* to only one *area*) Join the cell to the *connected area* and *redefine* that *area*.
(case: *connected* to more than one *area*) Join the cell to a suitable *area* and *merge* other *areas* to that *area*.

step 8: If there are any unprocessed cells, go to step 3.

The definitions of an *area*, *connectedness*, *redefinition* of an *area*, and *merging* are as follows. These definitions are

convinient for computation and require less memory space.

area: A hyperrectangle. Its attributes are a "record" and a status of either a *cluster* or a *candidate*. Each *area* is registered in an area table.

connectedness: 8-neighbours or 4-neighbours.

redefinition of an *area:* (a) Extend an *area* if and only if the *area* does not intersect with any other *areas* after extention of the *area*, and the ratio of the number of cells included in the *area* to the *area's* volume is greater than the threshold **R**. (b) Otherwise, make a new adjoining *area* whose attributes are the same as those of that *area*.

merging: (a) Between *candidates* or between a *candidate* and a *cluster*, make them equal in their attributes. (b) Between *clusters*, inhibit *merging*.

The threshold **N** of valley depths and the threshold **R** of the ratio of cells to volume are the major parameters in this method. Since **R** controls the fidelity in the extention of a *cluster's area* as a set of hyperrectangles, the cluster's form becomes more conscientious as **R** is closer to 1. In such a case, however, either a very large memory space is to be provided, or this algorithm is to stop at an earlier stage because of the limited length of the area table. As for the threshold **N**, **N** indicates the amount of sampling error. An unsuitably small **N** leads to too many *clusters*, and, vice versa, a very large **N** leads to too few *clusters*. The appropriate values of **R** and **N** have to be determined in accordance with the application field.

Table 2 illustrates the good performance of this clustering method for the synthetic data, which have 20,000 samples from a 10 class mixture in 4 dimensional space using a Gaussian random number generator and a random cluster generator. Fig. 3 is an example applying this clustering method to a LANDSAT scene ID.1145-00542. In both examples, the MDH is constructed by the multi-layered hashing scheme described in section III.

V. CONCLUSION

For a huge satellite image, this paper has described a new clustering method using local maximums of an MDH and an efficient histogram construction method using multi-layered histogram tables. Results are presented which show the utility and efficiency of these methods. The optimal values for **R** and **N** for each application field are items for future study. A quantitative evaluation of the clustering results is currently under way. These methods can be applied to classical clustering methods. At NAL these methods have been combined with an ISODATA-like method for forestry analysis.

VI. REFERENCES

1. R.O. Duda and P.E. Hart: Pattern Classification and Scene Analysis. John Wiley & Sons, 1973.
2. S. Shlien: Practical Aspects Related to Automated Classification of LANDSAT Imagery Using Lookup Tables. CCRS Report 75-2, 1975.
3. M. Goldberg and S. Shlien: A Clustering Scheme for Multispectral Images. IEEE SMC-8, No.2, pp86–92, 1978.

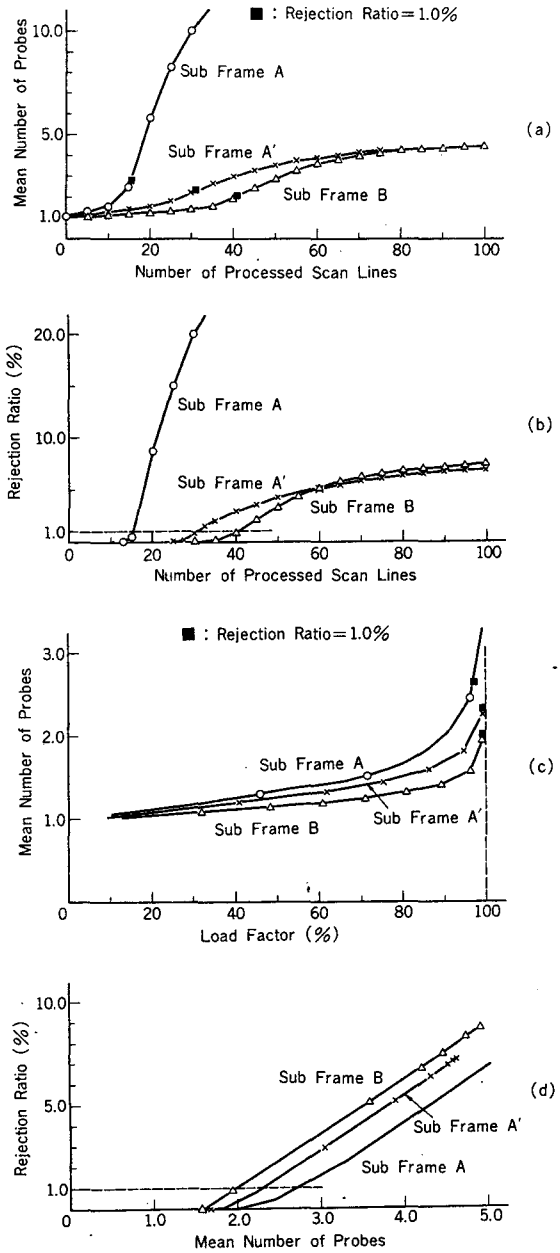

Fig. 1 Measurement of efficiency in basic hashing scheme.

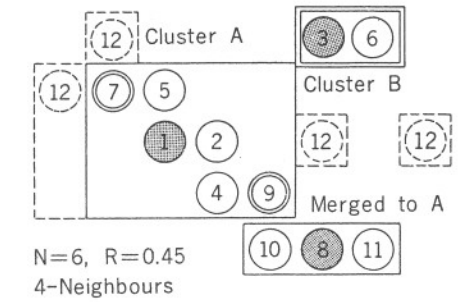

Fig. 2 Example of MDH clustering
1~12 shows the order after step 1.

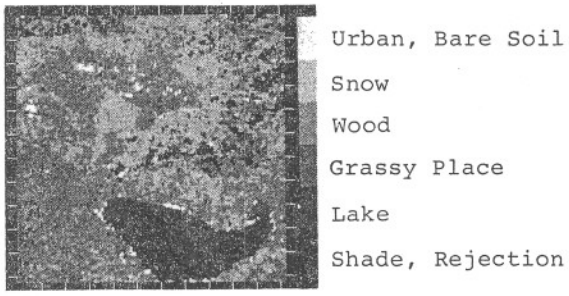

Fig. 3 MDH clustering result of LANDSAT ID.1145-00542. The area is around the Lake Yamanaka.

Table 1. Performance in MDH construction for a full scene of a LANDSAT ID.1145-00542.

	[Basic Scheme]		[Multi-Layered Scheme]	
	Gathering Ratio (%)	APU Time (sec)	Gathering Ratio (%)	APU Time (sec)
Step 1	65.84	1874.71	–	321.11
Step 2	–	–	89.57	447.83
Step 3	–	–	91.03	304.39
Total	65.84	1874.71	91.03	1073.33

No. of generated Layers : 46
Condition in Step 1 : Rejected Samples' Ratio = 1%
Table length : 20,000

Table 2 Performance of MDH clustering for synthetic data.

Cluster No.	Dist.	Clustering Error (%)	Cluster No.	Dist.	Clustering Error (%)
1	0.0146	0.49	6	0.0162	-0.60
2	0.0113	-1.10	7	0.0709	-5.93
3	0.1906	-13.46	8	0.1090	5.70
4	0.0051	-0.63	9	0.0285	0.22
5	0.2861	7.55	10	0.0518	5.19

Dist. : Separability Measure
Clustering Parameter : R = 0.001, N = 2

Kohtaro Matsumoto received both the B.S. degree and the M.S. degree from the Department of Mathematical Engineering and Instrumentation Physics of the University of Tokyo. His research interests include image processing, parallel processing, and parallel processor. He is on the research staff of the Computer Center of the National Aerospace Laboratory in Tokyo, Japan. Presently, he is working on developing the parallel algorithms for the image processing of a satellite image, and on its applications for forestry.

Masao Naka received the B.S. degree in Electrical Engineering from Ehime University, 1959 and joined NAL in 1959. From 1969 to 1970, as a NASA Fellow, he was with the Computer Science Research Laboratory at the University of Arizona, Tucson, Az., working on digital signal processing and computer control system. He is a section chief of the Data Processing Section at NAL's Computer Center. He has been involved with computer processing of a satellite image in such areas as data pre-processing, data analysis, and application to forestry, since 1976.

Hiromichi Yamamoto is a senior researcher at the Computer Center of the National Aerospace Laboratory in Tokyo. In 1978 he joined the Computer Vision Laboratory of the Computer Science Center at the University of Maryland as a visiting researcher and worked on pixel classification of multi-dimensional image data by use of the relaxation labeling scheme. His recent research areas include remote sensing, image processing, and computer science.