

รายการอ้างอิง



Bendiktsson J. A., Swain P. H., and Ersoy O. K. ,1990 , "*Neural network approaches versus statistical methods on classification of multisource remote sensing data*" , IEEE Transaction Geoscience and Remote Sensing, Vol. 28 , No.4, pp. 540-552.

Bichof and Leonardis A. ,1998, "*Finding Optimal Neural Network for Land Use Classification*", IEEE Transaction on Geoscience and Remote Sensing, Vol.36, No.1, January, pp.337-341

Bischof H. , Schneider W., and Pinz A.J. , 1992, "*Multispectral Classification of Landsat Images Using Neural Networks*" , IEEE Transactions on Geoscience and Remote Sensing, Vol. 30 , No.3, pp. 482-490.

Burks T.F., Shearer S.A. , Sobolik C.J., Fulton J.P. , Higgin S.F. , Thomasson J.A., Mueller T.G. and Samson S., 2000 "*Yield Prediction Using A Neural Network Classifier Trained Using Soil Landscape Features and Soil Fertility Data*",
<http://www.bae.uky.edu/~precag/PrecisionAg/Reports/n_network.htm>

Chong C.C. ,Jia J.C. and Mital D.P. , 1994 , "*Classification Of Multispectral Images Using BP-Neural Network Classifier-Input Codings Assessment*", School of Electronic Engineering, Nanyang Technological University Nanyang Avenue, Singapore 2263.

Congalton R. G. and Kass G., 1999 , Assessing the Accuracy of Remotely Sensed Data : Principles and Practices, Lewis Publishers

Gonzalez R.C. , 1993, Woods R.E., Digital Image Processing , Addison-Wesley Publishing Company, Inc.

Haykin S. ,1999 , *Neural Networks : a comprehensive foundation* , Prntice-Hall, Inc. ,New Jersey.

Heermann D.P. and Khazenie N., 1992, "*Classification of Multispectral Remote Sensing Data Using a Back-Propagation Neural Network*", IEEE Transactions on Geoscience and Remote Sensing, Vol.30, No.1, pp. 81-88.

Hepner G.F. , Logan T. , Ritten N., and Bryant N., 1990 , "*Artificial Neural Network Classification Using a Minimal Training Set : Comparison to Conventional Supervised Classification*" , Photogrammetric Engineering and Remote Sensing, Vol 56, No.4, pp.469-473.

International Journal of Remote Sensing, 1997, Special Issue on Neural Networks in Remote Sensing, Vol.18, no.4

Kamata S.I., Eason R.O. , Perez Arnulfo and Kawaguchi Eiji, 1992 , "*A Neural Network Classifier for LANDSAT Image Data.*" , Dept. of computer Engr., Kyushu Institute of Technology Tobata, Kitakyushu, Japan 804

Lillesand T. M. and Kiefer R. W. , 2000 , *Remote Sensing and Image Interpretation*, John Willey & Sons , Inc.

Lippmann R. P. , 1989, "*Pattern Classification Using Neural Networks*" , IEEE Communications magazine, November , pp.47-64

McClellan G.E., Dewitt R.N. , Hemmer T.H. , and others, 1989, "*Multispectral Image Processin With A Three-Layer Backpropagation Network*" , Proceedings of the International Joint Conference on Neural Networks, Washington D.C. , Vol.1 , pp.151-153.

Neural Network for Remote Sensing Application , IEEE Transaction on Geoscience and Remote Sensing , Vol.32 , No.5, pp.1096-2002

NIST/SEMATECH e-Handbook of Statistical Methods,
<http://www.itl.nist.gov/div898/handbook/>, 2001.

Paola J. D. and R. A. Schowengerdt , 1995, "*A reviews and analysis of backpropagation neural networks for classification of remotely-sensed multi-spectral imagery*" , International Journal of Remote Sensing , Vol.16, no.16, pp.3033-3058

Paola J.D. and Schowengerdt R.A. , 1995, "*A Detailed Comparison of Backpropagation Neural Network and Maximum-Likelihood Classifiers for Urban Land Use Classification*", IEEE Transactions on Geoscience and Remote Sensing, Vol.33, No.4, pp. 981-996.

Peter G. Z. , 2000, "*Neural Networks for Classification : A Survey*" , IEEE Transaction on Systems , Man, and Cybernetics-Part C: Application and Review, vol. 30, No. 4, pp451-462

Riedmiller M. and Braun H. ,1993, "*A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm*", Proc. Of the IEEE Int. Conf. On Neural Networks, pp. 586-591

Sarle W.S., ed. (1997), Neural Network FAQ, pat1 of 7: Introduction, periodic posting to the Usenet newsgroup comp.ai.neural-nets,URL:<ftp://ftp.sas.com/pub/neural/FAQ.html>

Skidmore A.K., Turner B.J., Brinkhof W. , and Knowles E., 1997, "*Performance of a Neural Network:Mapping Forest Using GIS and Remotly Sense Data*", Photogrammetric Engineering & Remote Sensing. Vol.63 , No.5, pp.501-514.

Sucharita G. and Manfred F., 1996, "*A Comparison of Three Neural Network Classifier for Remote Sensing Classification*", Dept. of Geography Boston University
Boston MA 02215.

Swain P. H. and Davis S. M. , 1978, *Remote Sensing : The Quantitative Approach*,
Mc Graw-Hill, Inc.

Trisirisatayawong I., 1994, "*Automatic Feature Extraction From Scanned Topographic Maps*", Doctor of Philosophy, Department of Surveying and Land
Information, The University of Melbourne.

Tzeng Y.C. , Chen K.S. and Member ,IEEE, Kao W.L. , and Fung A.K., 1994 "*A Dynamic Learning*

Yoshida T. and S. Omatu, 1994, *IEEE Transaction on Geoscience and Remote Sensing* ,
Vol.32 , No.5, pp.1103-1109

Zurada J.M., 1992, *Introduction to Artificial Neural Systems* , Block113 Eunos
Avenue3,#07-03 Gordon Industrial Building, Singapore : West Publishing .

ภาคผนวก

ภาคผนวก ก.

การจำแนกภาพดาวเทียมด้วยวิธีความคล้ายคลึงมากที่สุดและวิธีโครงข่ายใยประสาทเทียมโดยใช้ซอฟต์แวร์ ERDAS IMAGINE 8.5 และ MATLAB 6.0

เนื่องจากในปัจจุบัน ซอฟต์แวร์ทางด้าน การสำรวจระยะไกลที่ใช้ในการจำแนกภาพดาวเทียมส่วนใหญ่ ยังไม่ปรากฏเครื่องมือในการจำแนกภาพดาวเทียมด้วยวิธีโครงข่ายใยประสาทเทียม ดังนั้นในงานวิจัยครั้งนี้จึงได้เลือกใช้ซอฟต์แวร์ ERDAS IMAGINE 8.5 และ MATLAB 6.0 ในการจำแนกภาพดาวเทียมด้วยวิธีความคล้ายคลึงมากที่สุดและวิธีโครงข่ายใยประสาทเทียมเพื่อคำนวณหาความถูกต้องของการจำแนกและทำการจำแนกภาพดาวเทียมในพื้นที่ศึกษา ซึ่งในบทนี้จะกล่าวถึงส่วนประกอบ และรายละเอียดของขั้นตอนที่ใช้ในการจำแนกภาพดาวเทียมที่จัดทำขึ้นเพื่อใช้ในงานวิจัยครั้งนี้

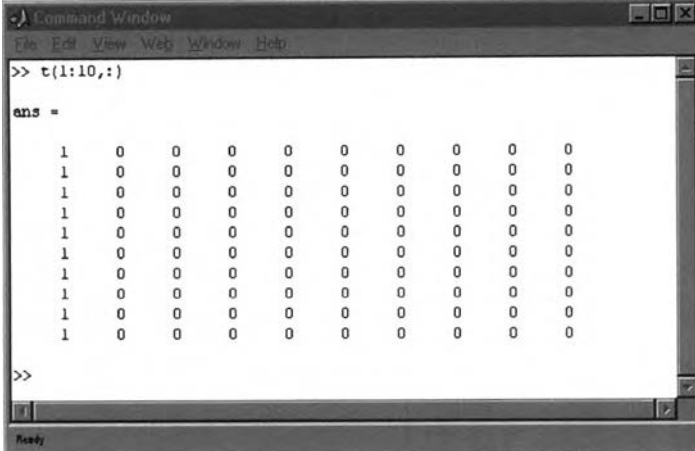
1.1 การจัดเตรียมข้อมูล

1.1.1 ข้อมูลตัวอย่างและข้อมูลทดสอบ

ข้อมูลที่ใช้สำหรับการเปรียบเทียบการจำแนกซึ่งประกอบด้วย ข้อมูลตัวอย่าง และข้อมูลทดสอบ ที่ได้จากการสำรวจข้อเท็จจริงในสนามแล้ว จะถูกทำการแปลงให้อยู่ในรูปแบบ ASCII ซึ่งสามารถใช้ฟังก์ชันที่อยู่ในซอฟต์แวร์ ERDAS IMAGINE 8.5 โดยจะแบ่งข้อมูลที่ได้เป็นสองส่วนคือข้อมูลตัวอย่าง ที่ใช้สำหรับการสร้างรูปแบบการจำแนกในแต่ละวิธี และข้อมูลทดสอบ ที่ใช้ในการวัดความถูกต้องที่ได้ในการจำแนก

1.1.2 ข้อมูลที่ใช้กำหนดผลลัพธ์

ผลลัพธ์ที่เป็นชนิดการจำแนกของข้อมูลจุดภาพที่ทราบแน่นอนจากการสำรวจข้อเท็จจริงในสนาม ที่ใช้ในงานวิจัยครั้งนี้จะถูกแสดงในรูปเมตริกซ์ โดยกำหนดให้มีจำนวนหลักของเมตริกซ์เท่ากับชนิดการจำแนก และจำนวนแถวของเมตริกซ์เท่ากับจำนวนของจุดภาพ ซึ่งสมาชิกที่แสดงชนิดการจำแนกของจุดภาพมีค่าเท่ากับ 1 ในขณะที่สมาชิกที่ไม่ใช่ผลลัพธ์มีค่าเท่ากับ 0 ดังแสดงตัวอย่างในรูปที่ ก-1



```

Command Window
File Edit View Web Window Help
>> t(1:10,:)
ans =
     1     0     0     0     0     0     0     0     0     0
     1     0     0     0     0     0     0     0     0     0
     1     0     0     0     0     0     0     0     0     0
     1     0     0     0     0     0     0     0     0     0
     1     0     0     0     0     0     0     0     0     0
     1     0     0     0     0     0     0     0     0     0
     1     0     0     0     0     0     0     0     0     0
     1     0     0     0     0     0     0     0     0     0
     1     0     0     0     0     0     0     0     0     0
     1     0     0     0     0     0     0     0     0     0
>>
Ready

```

รูปที่ ก-1 ตัวอย่างเมตริกซ์ผลลัพธ์ของข้อมูลตัวอย่างและข้อมูลทดสอบ

จากรูปที่ ก-1 แสดงตัวอย่างผลลัพธ์ของข้อมูลจุดภาพจำนวน 10 จุดภาพ ที่พบว่าเป็นชนิดการจำแนกชนิดที่ 1 หลังจากได้ทำการสำรวจข้อเท็จจริงในสนามแล้ว โดยกำหนดให้ในแต่ละแถวแทนผลลัพธ์ของแต่ละจุดภาพ และสมาชิกของเมตริกซ์ในหลักที่ 1 มีค่าเท่ากับ 1 เนื่องจากจุดภาพเป็นชนิดการจำแนกชนิดที่ 1

1.2 การจำแนกด้วยวิธีความคล้ายคลึงมากที่สุด

1.2.1 การสร้างรูปแบบการจำแนกด้วยวิธีความคล้ายคลึงมากที่สุด

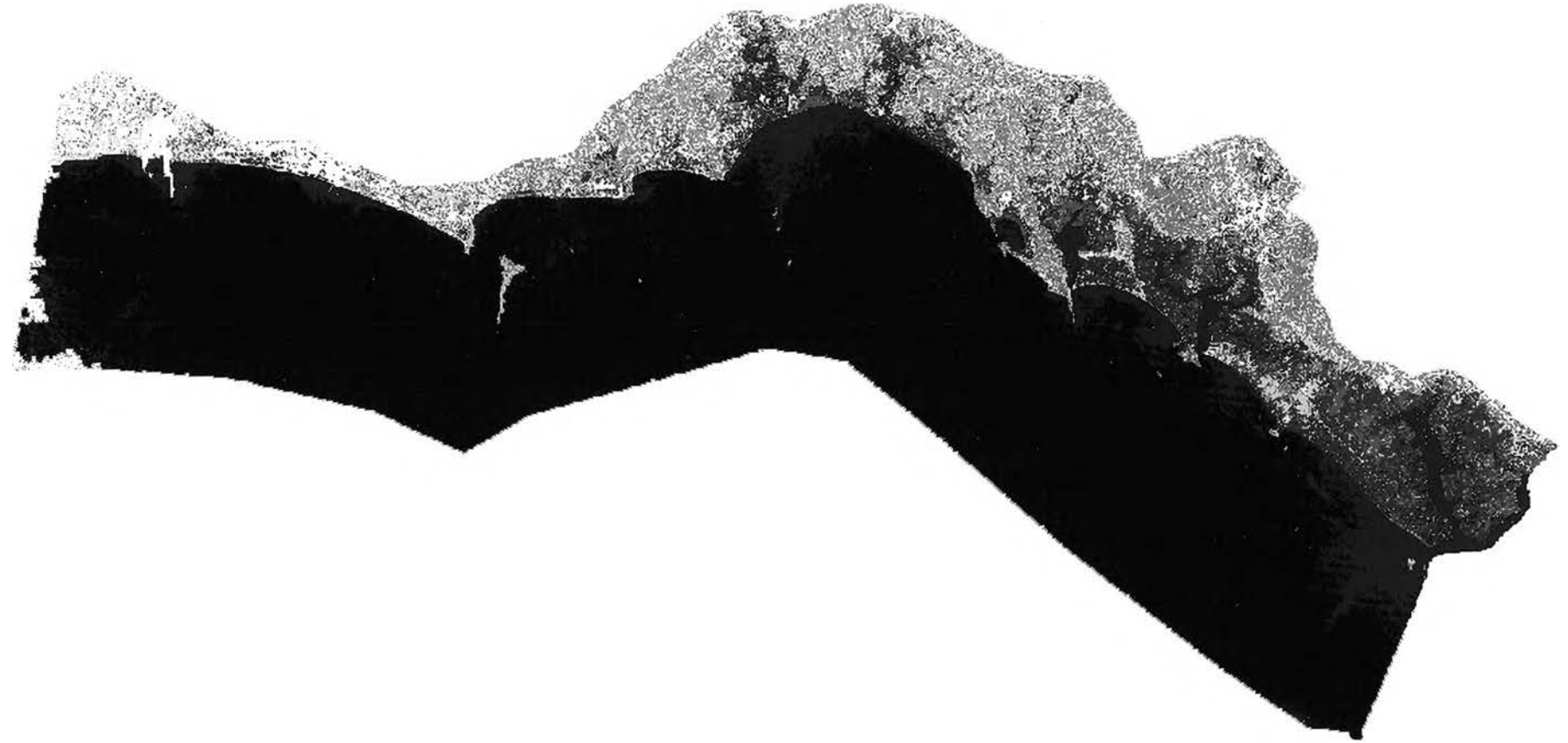
ข้อมูลตัวอย่างที่อยู่ในรูปแบบ ASCII จะถูกทำการหาค่าพารามิเตอร์ซึ่งเป็น Spectral Signature ของข้อมูล โดยผ่านโปรแกรมที่สร้างขึ้นในซอฟต์แวร์ MATLAB 6.0 ชื่อว่า mlhtrain.m ซึ่งผลลัพธ์ที่ได้จะเป็นชุดเมตริกซ์ของค่าเฉลี่ย ค่าความแปรปรวน ค่าความเบี่ยงเบนมาตรฐาน ค่าพิสัย ค่า Skewness และค่า Separability ของข้อมูลตัวอย่าง รวมถึงผลลัพธ์ของการจำแนกข้อมูลทดสอบที่อยู่ในรูป Confusion Matrix ที่ได้จากการจำแนกข้อมูลทดสอบโดยใช้ Spectral Signature ของข้อมูลตัวอย่าง

1.2.2 การจำแนกภาพดาวเทียมด้วยวิธีความคล้ายคลึงมากที่สุด

หลังจากได้ค่า Signature ที่คำนวณจากข้อมูลตัวอย่างแล้ว ค่าพารามิเตอร์ที่อยู่ในรูปของเมตริกซ์ จะถูกบันทึกอยู่ในรูปแบบของ MAT File ในซอฟต์แวร์ MATLAB 6.0 ชื่อไฟล์

signature.mat ซึ่งค่า Signature จะถูกนำมาใช้ในการคำนวณค่าความน่าจะเป็นของจุดภาพในการจำแนกภาพดาวเทียมโดยวิธีความคล้ายคลึงมากที่สุด

ในขั้นตอนของการจำแนกภาพดาวเทียม ข้อมูลของจุดภาพจะต้องถูกทำการแปลงให้อยู่ในรูปของ ASCII โดยใช้ฟังก์ชัน Convert Pixel to ASCII ในซอฟต์แวร์ ERDAS IMAGINE 8.5 จากนั้นจึงใช้โปรแกรมที่สร้างขึ้นในซอฟต์แวร์ MATLAB 6.0 ชื่อ mlhcls.m ซึ่งจะมีหน้าที่ในการจำแนกข้อมูลจุดภาพทีละจุดภาพโดยอาศัยการคำนวณค่าความน่าจะเป็นของแต่ละชนิดการจำแนกจากค่า Signature ของข้อมูลตัวอย่าง ซึ่งผลลัพธ์ที่ได้จะอยู่ในรูปของ ASCII File ชื่อว่า mlh_result.asc โดยผลลัพธ์ที่ได้จะยังคงเก็บค่าพิกัดของจุดภาพเอาไว้ ซึ่งสามารถแปลงให้อยู่ในรูปของ Thematic Map โดยใช้ฟังก์ชัน Convert ASCII to Pixel ในซอฟต์แวร์ ERDAS IMAGINE 8.5 โดยตัวอย่างผลลัพธ์ของการจำแนกที่ได้ในการจำแนกจุดภาพที่อยู่ในพื้นที่ศึกษาของงานวิจัยครั้งนี้ แสดงในรูปที่ ก-2



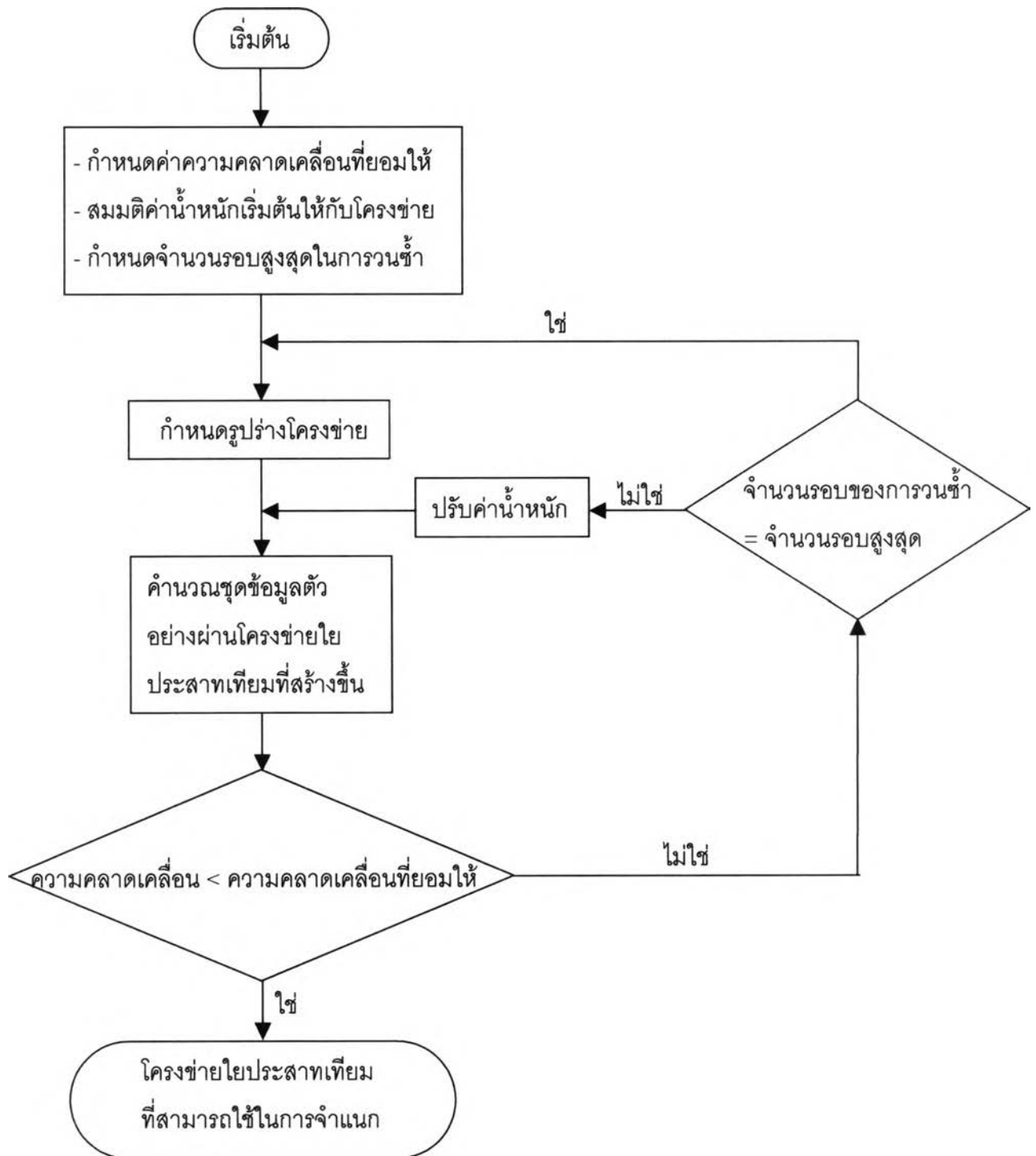
รูปที่ ก-2 ผลการจำแนกจุดภาพในพื้นที่ศึกษาด้วยวิธีความคล้ายคลึงมากที่สุด

1.3 การจำแนกด้วยวิธีโครงข่ายประสาทเทียม

1.3.1 การสร้างโครงข่ายเพื่อใช้ในการจำแนกด้วยวิธีโครงข่ายประสาทเทียม

ขั้นตอนแรกของการจำแนกภาพดาวเทียมด้วยวิธีโครงข่ายประสาทเทียม หลังจากที่ได้เตรียมข้อมูลตัวอย่าง ข้อมูลทดสอบ และผลลัพธ์ ที่อยู่ในรูปของเมตริกซ์ คือการสร้างโครงข่ายประสาทเทียม ที่มีคุณสมบัติในการจำแนกข้อมูลตัวอย่างได้อย่างถูกต้อง ซึ่งได้มาจากกระบวนการปรับค่าน้ำหนักภายในโครงข่ายประสาทเทียม โดยวิธีการสร้างโครงข่ายประสาทเทียมอธิบายตามรูปที่ ก-3

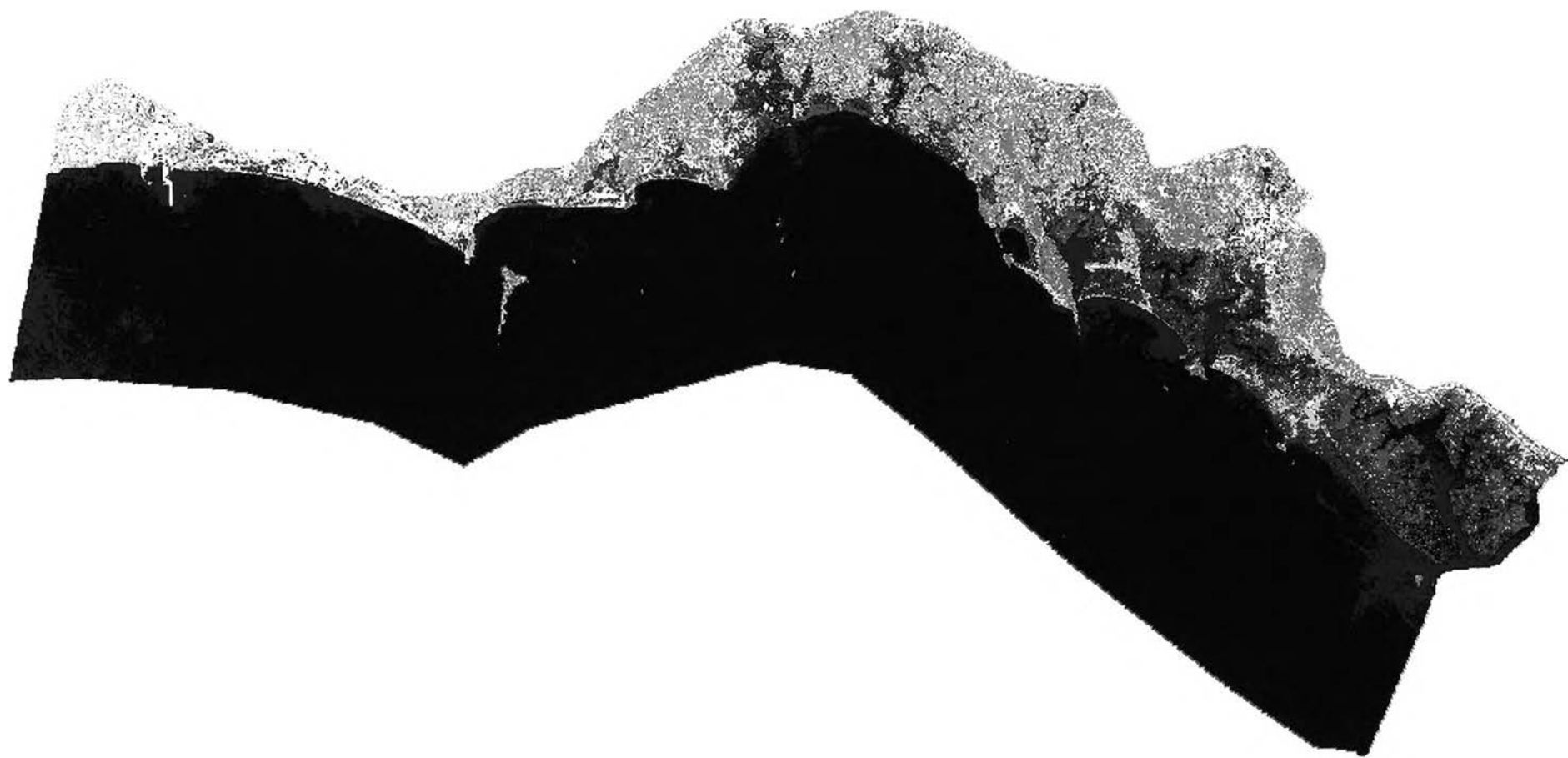
ในงานวิจัยครั้งนี้ได้สร้างโปรแกรมที่ใช้ในการสร้างโครงข่ายประสาทเทียมตามกระบวนการในรูปที่ ก-3 ในซอฟต์แวร์ MATLAB 6.0 ชื่อว่า nntrain.m มีคุณสมบัติในการปรับค่าน้ำหนักของโครงข่าย โดยรูปร่างของโครงข่าย ค่าความคลาดเคลื่อนที่ยอมรับได้ และจำนวนรอบในการวนซ้ำมากที่สุด จะต้องทำการกำหนดก่อนทำการคำนวณแบบวนซ้ำเพื่อหาค่าน้ำหนักที่เหมาะสม ซึ่งผลลัพธ์ที่ได้จะเป็นชุดเมตริกซ์ที่มีคุณสมบัติในการจำแนกข้อมูลตัวอย่างได้ถูกต้อง



รูปที่ ก-3 กระบวนการสร้างโครงข่ายใยประสาทเทียมเพื่อใช้ในการจำแนกภาพดาวเทียม

1.3.2 การจำแนกภาพดาวเทียมด้วยวิธีโครงข่ายประสาทเทียม

การจำแนกภาพดาวเทียมด้วยวิธีโครงข่ายประสาทเทียม ทำการจำแนกผ่านโปรแกรมที่เขียนขึ้นในซอฟต์แวร์ MATLAB 6.0 ชื่อว่า nncls.m ซึ่งมีคุณสมบัติในการคำนวณค่าความสว่างของจุดภาพผ่านโครงข่ายประสาทเทียมที่สร้างขึ้น ที่ละจุดภาพจนท้ายสุดให้ผลลัพธ์ของการจำแนกอยู่ในรูปของ ASCII โดยจุดภาพยังคงมีค่าพิกัดของจุดภาพก่อนทำการจำแนก ซึ่งจะสามารถถูกแปลงให้อยู่ในรูปของ Thematic map โดยใช้ฟังก์ชัน Convert ASCII to Pixel ในซอฟต์แวร์ ERDAS IMAGINE 8.5 ผลลัพธ์ที่ได้จากการจำแนกจุดภาพในพื้นที่ศึกษาด้วยวิธีโครงข่ายประสาทเทียมแสดงในรูปที่ ก-4



รูปที่ ก-4 ผลการจำแนกจุดภาพในพื้นที่ศึกษาด้วยวิธีโครงข่ายใยประสาทเทียม

ภาคผนวก ข.

โปรแกรมที่สร้างขึ้นเพื่อใช้ในงานวิจัย

ในงานวิจัยครั้งนี้โปรแกรมที่สร้างขึ้นเพื่อใช้ในการจำแนกด้วยวิธีความคล้ายคลึงมากที่สุด และวิธีโครงข่ายประสาทเทียม สร้างขึ้นจากซอฟต์แวร์ MATLAB 6.0 ซึ่งแต่ละโปรแกรมมีคุณสมบัติและองค์ประกอบดังนี้

1. teacher.m

เป็นโปรแกรมที่ใช้ในขั้นตอนการเตรียมข้อมูลของผลลัพธ์เพื่อใช้ในการจำแนก โดยมีค่าที่จะต้องกำหนดให้กับโปรแกรมได้แก่ สมาชิกในเมตริกซ์ dd ซึ่งเป็นจำนวนของข้อมูลตัวอย่าง และข้อมูลทดสอบในแต่ละชนิดของการจำแนก และขนาดของเมตริกซ์ t ซึ่งมีขนาดของแถวเท่ากับจำนวนจุดภาพทั้งหมดที่เป็นข้อมูลตัวอย่างและข้อมูลทดสอบ และขนาดของหลักเท่ากับจำนวนของชนิดการจำแนก

```
%teacher.m
```

```
dd=[84 94 120 72 91 103 96 91 71 94 70 81 109 79 77 131 119 119 66 99];
```

```
t=zeros(1866,10);
```

```
init=1;
```

```
initt=sum(dd(1:10))+1;
```

```
for i = 1:10
```

```
    t(init:(init+dd(i)-1),i)=1;
```

```
    t(initt:(initt+dd(10+i)-1),i)=1;
```

```
    init=init+dd(i);
```

```
initt=initt+dd(10+i);
```

```
end
```

2. mlhtrain.m

มีหน้าที่ในการคำนวณค่า Spectral Signature ของข้อมูลตัวอย่าง รวมทั้งคำนวณค่าพารามิเตอร์ทางสถิติอย่างอื่นเช่น ค่า Skewness และค่า Separability และนำค่าพารามิเตอร์ต่างๆ ที่ได้มาทำการจำแนกข้อมูลทดสอบเพื่อคำนวณหาความถูกต้องในการจำแนกรูปของ Confusion Matrix

```
% Maximumlikelihood
% m is number of data per class
% type is number of category
% data is array of data
load Simulate
p = p';
t = t';
[rp cp] = size(p);
[rt ct] = size(t);

% Make the data to follow to class order.
disp('Please, Enter sequence of data')
seq = input('Enter : ');

% Input the number of data for each class.
m = zeros(ct,1);
for i = 1:ct
    disp('Please, Enter number of each class')
    m(i) = input('Enter : ');
end
```

```
% Define the data set.
train = [1:916];
test = [917:1866];
%test = [1:916];
ptrn = p(train,:);
ptst = p(test,:);

ttrn = t(train,:);
ttst = t(test,:);

n = cp;
x = ptrn(seq,:);
y = ptst;

testset = ttst;
mn = zeros(ct,cp);
mdn = zeros(ct,cp);
rng = zeros(ct,cp);
mmx = zeros(ct,cp);
mmn = zeros(ct,cp);
skw = zeros(ct,cp);
sd = zeros(ct,cp);
cv = zeros(cp,cp,ct);
ccd = zeros(ct,3);

[l c] = size(y);

sas=0;
unsas=0;
tt = cputime;
```



```

% Compute the signature.
tmp = 0;
for i = 1:ct
    figure(i)
    temp = [tmp+1:tmp+m(i)];
    mn(i,:) = mean(ptrn(temp,:));
    mdn(i,:) = median(ptrn(temp,:));
    rng(i,:) = range(ptrn(temp,:));
    cv(:,i) = cov(ptrn(temp,:));
    ccv = cv(:,i);
    for j = 1:cp
        subplot(cp,1,j);
        hist(ptrn((temp(1):temp(m(i))),j))
        axis([0 255 0 22])
        Title('Histogram')
        xlabel('DN')
        mmx(i,j)= max(ptrn((temp(1):temp(m(i))),j));
        mmn(i,j)= min(ptrn((temp(1):temp(m(i))),j));
        sd(i,j) = ccv(j,j);
        sd(i,j) = sqrt(sd(i,j));
    end
    tmp = tmp+m(i);
end

figure(ct+1)
for i = 1:cp
    for j = 1:ct
        ccd(j,1) = mmn(j,i);
        ccd(j,2) = rng(j,i);
        ccd(j,3) = 255-(rng(j,i)+mmn(j,i));
    end
end

```

```

subplot(cp/2,2,i);
Title('Coincident spectral plots')
barh(ccd,'stack')
axis([0 255 0 12])
xlabel('DN')
ylabel('Class')
grid on
end

for i = 1:ct
    for j=1:cp
         $skw(i,j) = (mn(i,j)-mdn(i,j)) / sd(i,j);$ 
    end
end

result = zeros(l,ct);
mx = zeros(l,1);
sol= zeros(l,1);

% Classify test set.
for ii = 1:l
    seed = y(ii,:);
    seed = seed';
    for jj = 1:ct
        result(ii,jj) = ml(mn(jj,:),cv(:,jj),seed);
    end
end

tm = cputime - tt;

for i = 1:l

```

```

    mx(i) = max(result(i,:));
end

for h = 1:l
    for g = 1:ct
        if mx(h)== result(h,g)
            sol(h) = g;
        end
    end
end

end

%display(' result of each pixel for each class ')
result;
%display(' result of classification ' )
sol;
[ro,co]=size(testset);
check = zeros(ro,1);
for i =1:ro
    for j =1:co
        if testset(i,j)==1
            check(i)=j;
        end
    end
end

end

for i = 1:ro
    if sol(i)==check(i)
        sas = sas+1;
    else
        unsas = unsas+1;
    end
end

end

```

```

['Accuracy of Classification is :',num2str((sas/(sas+unsas))*100),' Percent']
%disp('Percent accuracy')
%disp((sas/(sas+unsas))*100)
%disp('Cpu Time :')
%tm
%disp('Second')
count = zeros(co,1);
for i = 1:co
    for j = 1:ro
        if check(j) == i
            count(i)= count(i)+1;
        end
    end
end
end

keep = 0;
j = 1;
for i = 1:co
    %figure(i);
    figure(ct+2)
    keep = keep+count(i);
    pos = [j:keep];
    y = sol(pos);
    z = check(pos);
    j = keep+1;
    subplot(2,ct/2,i);
    plot(pos,y,'r:O',pos,z,'b:.');
    %legend('Maximumlikelihood','Ground Truth')
    xlabel('Number of Testing data')
    ylabel('Number of Class')
    %Title('Checking Result of Classification')

```

```

    grid on
end
conf = zeros(ct,ct+3);
for i = 1:ro
    j = sol(i);
    k = check(i);
    conf(k,j) = conf(k,j) + 1;
end

for i = 1:ct
    conf(i,ct+1)=conf(i,i);
    conf(i,ct+2)=sum(conf(i,1:ct));
    conf(i,ct+3)=100*(conf(i,i)/conf(i,ct+2));
end

dlmwrite('mean',mn);
spr = zeros(ct,ct);
mn = mn';
for i = 1:ct
    cvi = cv(:,i);
    for j = 1:ct
        cvj = cv(:,j);
        spr(i,j) = 0.5 * trace((cvi-cvj)*(inv(cvj)-
inv(cvi)))+0.5*trace((inv(cvi)+inv(cvj))*(mn(:,i)-mn(:,j))*(mn(:,i)-mn(:,j))));
        spr(i,j) = 2000*(1-(exp(-(spr(i,j))/8)));
    end
end

dlmwrite('median',mdn);
dlmwrite('std',sd);
dlmwrite('mlh_confusion',conf);

```

```
dlmwrite('min',mmn);
dlmwrite('max',mmx);
dlmwrite('skw',skw);
dlmwrite('separability',spr);
```

3. mlhcls.m

เป็นโปรแกรมที่สร้างขึ้นเพื่อใช้สำหรับการจำแนกภาพดาวเทียมโดยวิธีความคล้ายคลึงมากที่สุดจากค่า Spectral Signature ที่คำนวณจากข้อมูลตัวอย่าง ซึ่งผลที่ได้จากการจำแนกในรูปแบบของ ASCII จะอยู่ในไฟล์ mlh_result.asc

```
load Simulate
load signature
p = p';
t = t';
[rp cp] = size(p);
[rt ct] = size(t);
fid = fopen('swamp1.asc','r');
fidd = fopen('mlh_result.asc','w');
fprintf(fidd,' F1 : c:/chantaburi/utm_indian/geo_p128r51_03nov2000_indian.img\n\n
F1          \n      X      Y  B1\n');
tt=0;
while feof(fid) == 0
    tt=tt+1;
    tline = fgetl(fid);
    if tt>4
        tline=str2num(tline);
        %tline(10)=[];
        seed = tline;
        tline(3:7)=[];
        seed(1:2)= [];
        %seed(6) = [];
```

```

seed = seed';
cresult = zeros(rt,1);
for j = 1 : rt
    cresult(j,1) = mll(mn(:,j),cv(:,j),seed);
end
cmx = max(cresult);
for j = 1:rt
    if cmx == cresult(j);
        csol = j;
    end
end
tline = [tline,csol];
fprintf(fid,'%8.2d %8.2d %4.2d\n',tline);
['This time, 'num2str(tt-4),' pixels are classsified']
end
end
fclose('all');
display('Finished MLH Classify Process')

```

4. nntrain.m

เป็นโปรแกรมที่สร้างขึ้นสำหรับการสร้างโครงข่ายประสาทเทียม โดยอาศัยชุดข้อมูลตัวอย่างที่ทราบผลลัพธ์ที่แน่นอน

```

%Neural Network Classification for Satellite Image
load Simulate
sas = 0;
unsas = 0;
[pn,meanp,stdp,tn,meant,stdt] = prestd(p,t);
[R,Q] = size(p);
[x,y] = size(t);

```

```
%Define each set of data.

iitr = [1:916];
%iitst = [1:916];
iitst = [917:1866];
%ival = [];

%pclsfy = trastd(dat,meanp,stdp);
ptr = pn(:,iitr);
ttr = tn(:,iitr);
ptst = pn(:,iitst);
ttst = tn(:,iitst);
%val.P = pn(:,ival);
%val.T = tn(:,ival);

%setting all of parameter
net = newff(minmax(pn),[23 x],{'logsig','purelin'},'trainrp');
net.trainParam.epochs = 15000;
net.trainParam.mc = 0.7;
net.trainParam.show = 5;
net.trainParam.goal = 1e-2;
net = init(net);

%Learning step
tm = cputime;
%[net,tr] = train(net,ptr,ttr,[],[],val);
[net,tr] = train(net,ptr,ttr);
clsfytme = cputime - tm;
an = sim(net,ptst);
a= poststd(an,meant,stdt);
```



```

a = a';
[ro co] = size(a);
test = t(:,iitst);
test = test';
soll = zeros(ro,1);
chk = zeros(ro,1);

for i = 1:ro
    for j = 1:co
        if a(i,j) == max(a(i,:))
            soll(i)=j;
        end
        if test(i,j) == 1
            chk(i)=j;
        end
    end
end

%compute the accuracy
for i = 1:ro
    if soll(i)==chk(i)
        sas = sas+1;
    else
        unsas = unsas +1;
    end
end

%csoll = csoll';

%Output
['Accuracy of Classification is :',num2str((sas/(sas+unsas))*100),' Percent']
%(sas/(sas+unsas))*100

```

```

%disp('Percent')
soll = soll';
chk = chk';
['Training time is :',num2str(clsfytime),' seconds']
%disp('Training Time :')
%clsfytime
%disp('Seconds')

[co ro] = size(ttst);
count = zeros(co,1);
for i = 1:co
    for j = 1:ro
        if chk(j) == i
            count(i)= count(i)+1;
        end
    end
end

%Plot the result of classification.
count;
keep = 0;
j = 1;
figure(1)
for i = 1:co
    %figure(i+co)
    keep = keep+count(i);
    pos = [j:keep];
    y = soll(pos);
    z = chk(pos);
    j = keep+1;
    subplot(2,co/2,i);

```

```

plot(pos,y,'r:O',pos,z,'b:.'')
%legend('Neural Network','Ground Truth')
xlabel('Number of Testing data')
ylabel('Number of Class')
%Title('Checking Result of Classification')
grid on
end

```

```

% "conf" is confusionmatrix
soll = soll';
chk = chk';
conf = zeros(co,co+3);
for i = 1:ro
    j = soll(i);
    k = chk(i);
    conf(k,j) = conf(k,j) + 1;
end
for i = 1:co
    conf(i,co+1)=conf(i,i);
    conf(i,co+2)=sum(conf(i,1:co));
    conf(i,co+3)=100*(conf(i,i)/conf(i,co+2));
end
dlmwrite('NN_confusion',conf);
%dlmwrite('NN_solution',csoll);

```

5. nncls.m

เป็นโปรแกรมที่ใช้สำหรับการจำแนกข้อมูลด้วยวิธีโครงข่ายใยประสาทเทียมโดยอาศัยโครงข่ายที่ผ่านการปรับค่าน้ำหนัก จนมีคุณสมบัติในการจำแนกข้อมูลตัวอย่างได้อย่างถูกต้อง ผลลัพธ์ที่ได้จากการจำแนกในรูปแบบของ ASCII จะอยู่ในไฟล์ nn_result.asc

```

%Neural Network Classification for Satellite Image

load simulate

load trained_net

[R,Q] = size(p);

[x,y] = size(t);

fid = fopen('swamp1.asc','r');
fidd = fopen('nn_result.asc','w');

fprintf(fidd,' F1 : c:/chantaburi/utm_indian/geo_p128r51_03nov2000_indian.img\n\n
F1          \n      X      Y  B1\n');

tt=0;

while feof(fid) == 0
    tt=tt+1;
    tline = fgetl(fid);
    if tt>4
        tline=str2num(tline);
        %tline(10)=[];
        seed  = tline;
        tline(3:7)=[];
        seed(1:2)= [];
        %seed(6) = [];
        dat = seed';
        pclsfy = trastd(dat,meanp,stdp);
        can = sim(net,pclsfy);
        ca = poststd(can,meant,stdt);
        ca = ca';
        %[cro cco] = size(ca);
        mmx = max(ca);
        for j = 1:x
            if ca(j) == mmx
                csoll = j;
            end
        end
    end
end

```

```
        end
    end
    tline = [tline,csol];
    fprintf(fid,'%8.2d %8.2d %4.2d\n',tline);
    ['This time, 'num2str(tt-4),' pixels are clssified']
end
end
fclose('all');
display('Finished Neural Network Classify Process')
```

ประวัติผู้เขียนวิทยานิพนธ์

ผู้เขียนชื่อ นายชัชชัย เดชเกตู เกิดเมื่อวันที่ 5 กรกฎาคม พ.ศ.2520 ที่จังหวัด
กรุงเทพมหานคร สำเร็จการศึกษาปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมโยธาจาก
มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี เมื่อปี พ.ศ. 2542