APPENDIX A

Matlab Code for Binarization

```
X = imread('v1.jpg');
X = rgb2gray(X);
X = imresize(X,[32 32],'bicubic');

w_dim = 7;
nhood = ones(w_dim,w_dim);

min = ordfilt2(X, 1, nhood,'symmetric');
max = ordfilt2(X, w_dim*w_dim , nhood,'symmetric');

T = (min + max)./ 2;

X1 = (X > T);
imshow(X1)
[r,c]=size(X1);

f = fopen('ab.txt','w');
for i = 1 : r
    for j = 1 : c
        fprintf(f,'%d ',X1(i,j));
    end
    fprintf(f,'\r\n');
end
fclose(f);
```

APPENDIX B

Matlab Code for Longest Run Feature Generation

```
load ab.txt;
s = ab;
[r,c]=size(s);

sum = 0;
max_val=ones(1,r); index=ones(1,r);id=0;

for i=1:r
    f=0;count=0;max = 0;
    for j=1:c
        if(f==1 && s(i,j)==0)
            count = count + 1;
        elseif(s(i,j)==1)
            f=0;
            if(count > max)
                max = count ;
                count = 0;
                index(i) = id;
            end
        elseif(s(i,j) == 0)
            f=1;
            count = count + 1;
            id  = j;
        end
    end
    if(count>max)
        max = count ;
        index(i) = id;
    end
    max_val(i) = max;
end

for i = 1:r
    for j = index(i):c
        if(s(i,j)== 1)
```

```
            break;
        else
            s(i,j) = max_val(i);
        end
    end
end

[r,c] = size(max_val);
for i = 1 : c
    sum = sum + max_val(i);
end

f2 = fopen('v1_sum.txt','w');
fprintf(f2,'%d',sum);
fclose(f2);

f = fopen('v1_s.txt','w');
for i = 1 : 32
    for j = 1 : 32
        fprintf(f,'%d ',s(i,j));
    end
    fprintf(f,'\r\n');
end
fclose(f);

f1 = fopen('v1_m.txt','w');
for j = 1 : 32
     fprintf(f1,'%d ',max_val(j));
end
fclose(f1);
```

Matlab Code for Shadow Feature Generation

```
function [shadow1 shadow2] = diagonal_shadow(X,theta);

one_im = true(16,16);
mask = ~(imrotate(one_im,45));

Xr = imrotate(X,theta);
Xr = Xr | mask;
[m n] = size(Xr);

shadow1 = true(m,1);
shadow2 = true(m,1);

for i=1:n
     for j=1:ceil(m/2)
          if Xr(i,j)== false
               shadow1(i,1) = false;
               break;
          end
     end
end

for i=1:n
     for j=ceil(m/2)+1:m
          if Xr(i,j)== false
               shadow2(i,1) = false;
               break;
          end
     end
end

shadow1 = imresize(shadow1,[16 1],'bicubic');
shadow2 = imresize(shadow2,[16 1],'bicubic');

X = imread('c1.jpg');
X = rgb2gray(X);
X = imresize(X,[32 32],'bicubic');

w_dim = 7;
nhood = ones(w_dim,w_dim);

min = ordfilt2(X, 1, nhood,'symmetric');
max = ordfilt2(X, w_dim*w_dim , nhood,'symmetric');
```

```
T = (min + max)./ 2;

X1 = (X > T);

cro1 = imcrop(X1,[1 1  16 16]);
cro2 = imcrop(X1,[1 16 16 32]);
cro3 = imcrop(X1,[16 1 32 16]);
cro4 = imcrop(X1,[16 16 32 32]);

resA = false(16,4);

for i = 1:16
   for j = 1:i
       if cro1(i,j) == false
            resA(i,1) = true;
            break;
       end
   end
end

for i = 1:16
   for j = i:16
       if cro1(j,i) == false
            resA(i,2) = true;
            break;
       end
   end
end

for j = 1:16
   for i = j:16
       if cro1(i,j) == false
            resA(j,3) = true;
            break;
       end
   end
end

for j = 1:16
   for i = 1:j
       if cro1(j,i) == false
            resA(j,4) = true;
            break;
       end
   end
end
```

```
%figure 1
resA;

resB = false(16,4);

for i = 1:16
   for j = 1:i
       if cro2(i,j) == false
            resB(i,1) = true;
            break;
       end
   end
end

for i = 1:16
   for j = i:16
       if cro2(j,i) == false
            resB(i,2) = true;
            break;
       end
   end
end

for j = 1:16
   for i = j:16
       if cro2(i,j) == false
            resB(j,3) = true;
            break;
       end
   end
end

for j = 1:16
   for i = 1:j
       if cro2(j,i) == false
            resB(j,4) = true;
            break;
       end
   end
end

%figure 2
resB;

resC = false(16,4);

for i = 1:16
```

```
   for j = 1:i
       if cro3(i,j) == false
            resC(i,1) = true;
            break;
       end
   end
end

for i = 1:16
   for j = i:16
       if cro3(j,i) == false
            resC(i,2) = true;
            break;
       end
   end
end

for j = 1:16
   for i = j:16
       if cro3(i,j) == false
            resC(j,3) = true;
            break;
       end
   end
end

for j = 1:16
   for i = 1:j
       if cro3(j,i) == false
            resC(j,4) = true;
            break;
       end
   end
end

%figure 3
resC;

resD = false(16,4);

for i = 1:16
   for j = 1:i
       if cro4(i,j) == false
            resD(i,1) = true;
            break;
       end
   end
```

```
end

for i = 1:16
   for j = i:16
       if cro4(j,i) == false
            resD(i,2) = true;
            break;
       end
   end
end

for j = 1:16
   for i = j:16
       if cro4(i,j) == false
            resD(j,3) = true;
            break;
       end
   end
end

for j = 1:16
   for i = 1:j
       if cro4(j,i) == false
            resD(j,4) = true;
            break;
       end
   end
end

%figure 4
resD;

top_left = imcrop(X1, [1 1 15 15]);
top_right = imcrop(X1, [16 1 15 15]);
bottom_left = imcrop(X1, [1 16 15 15]);
bottom_right = imcrop(X1, [16 16 15 15]);

[sh1 sh2] = diagonal_shadow(top_left,-45);
[sh3 sh4] = diagonal_shadow(bottom_right,-45);
[sh5 sh6] = diagonal_shadow(top_right,45);
[sh7 sh8] = diagonal_shadow(bottom_left,45);

res=[resA resB resC resD]
sh = [sh1 sh2 sh3 sh4 sh5 sh6 sh7 sh8]
```

Matlab Code for Quadtree-based Feature Genaration

```
X = imread('c6.jpg');
X = rgb2gray(X);
X = imresize(X,[32 32],'bicubic');

w_dim = 7;
nhood = ones(w_dim,w_dim);

min = ordfilt2(X, 1, nhood,'symmetric');
max = ordfilt2(X, w_dim*w_dim , nhood,'symmetric');

T = (min + max)./ 2;
X1 = (X < T);

cro1 = imcrop(X1,[1 1  16 16]);
cro2 = imcrop(X1,[1 16 16 32]);
cro3 = imcrop(X1,[16 1 32 16]);
cro4 = imcrop(X1,[16 16 32 32]);

[r,c]=size(cro1);
r;
c;
sx1=0;
sy1=0;

for i=1:r
    for j=1:c
       sx1=sx1+i*cro1(i,j);
       sy1=sy1+j*cro1(i,j);
     end
end
sx1=sx1/(r*c)
sy1=sy1/(r*c)

[r,c]=size(cro2);
sx2=0;
sy2=0;

for i=1:r
    for j=1:c
       sx2=sx2+i*cro2(i,j);
       sy2=sy2+j*cro2(i,j);
     end
end
sx2=sx2/(r*c)
```

```
sy2=sy2/(r*c)

[r,c]=size(cro3);
sx3=0;
sy3=0;

for i=1:r
    for j=1:c
       sx3=sx3+i*cro3(i,j);
       sy3=sy3+j*cro3(i,j);
     end
end
sx3=sx3/(r*c);
sy3=sy3/(r*c);

[r,c]=size(cro2);
sx4=0;
sy4=0;

for i=1:r
    for j=1:c
       sx4=sx4+i*cro4(i,j);
       sy4=sy4+j*cro4(i,j);
     end
end

sx4=sx4/(r*c)
sy4=sy4/(r*c)

sx_sum = (sx1 + sx2 + sx3 + sx4)/4;
sy_sum = (sy1 + sy2 + sy3 + sy4)/4;

sx = [sx1 sx2 sx3 sx4 sx_sum];
sy = [sy1 sy2 sy3 sy4 sy_sum];

f = fopen('c6_q.txt','w');
fprintf(f,'%f %f ',sx,sy);
fclose(f);
```

APPENDIX C

Matlab Code for Neural Network

```
load v1_m.txt;
a = v1_m;
load v2_m.txt;
b = v2_m;
load v3_m.txt;
c = v3_m;
load v4_m.txt;
d = v4_m;
load v5_m.txt;
e = v5_m;
load v6_m.txt;
f = v6_m;
load v7_m.txt;
g = v7_m;
load v8_m.txt;
h = v8_m;
load v9_m.txt;
io = v9_m;
load v10_m.txt;
jo = v10_m;
load v11_m.txt;
ko = v11_m;

a1=[];
[ro1,cl1]=size(a);

for j=1:cl1
    a1=[a1;a(j)];
end

b1=[];
for j=1:cl1
     b1=[b1;b(j)];
end

c1=[];
for j=1:cl1
```

```
     c1=[c1;c(j)];
end
d1=[];
for j=1:cl1
     d1=[d1;d(j)];
end

e1=[];
for j=1:cl1
     e1=[e1;e(j)];
end

f1=[];
for j=1:cl1
     f1=[f1;f(j)];
end

g1=[];
for j=1:cl1
     g1=[g1;g(j)];
end

h1=[];
for j=1:cl1
     h1=[h1;h(j)];
end

i1=[];
for j=1:cl1
     i1=[i1;io(j)];
end

j1=[];
for j=1:cl1
     j1=[j1;jo(j)];
end

k1=[];
for j=1:cl1
     k1=[k1;ko(j)];
end

le=[a1 b1 c1 d1 e1 f1 g1 h1 i1 j1 k1];
t=eye(11);
net=newff(minmax(le),[50,11],{'tansig','purelin'},'traingd');
net.trainParam.epochs = 1000;
net.trainParam.goal=1e-2;
```

```
net.trainParam.lr=0.01;
net.trainParam.mc=0.1
net = train(net,le,t);
Y = sim(net,le);

cor=0;
for i=1:11
    index=1;
    for j=1:11
        if Y(j,i)>Y(index,i)
            index=i;
        end
    end
    if index==i
        cor=cor+1;
    end
end

'Train Cases Recognized'
cor

'Train Cases Recognition Accuracy'
cor/10*100

% Test Image

load v1_m.txt;
a = v1_m;
a1=[];
for j=1:32
   a1=[a1;a(j)];
end

match=sim(net,a1);

'Output vector for test image'
match
index=1;
for j=1:11
    if match(j)> match(index)
        index=j;
    end
end
'test image output'
index

[a,b]=size(match);
```

```
 f = fopen('lnv1.txt','w');
 for i = 1 : a
     for j = 1 : b
         fprintf(f,'%f ', match(i,j));
     end
     fprintf(f,'\r\n');
 end
 fclose(f);
```

```
%%Classifier of feature set 2

load v1_s.txt;

a = v1_s;

load v2_s.txt;

b = v2_s;

load v3_s.txt;

c = v3_s;

load v4_s.txt;

d = v4_s;

load v5_s.txt;

e = v5_s;

load v6_s.txt;

f = v6_s;

load v7_s.txt;

g = v7_s;

load v8_s.txt;

h = v8_s;

load v9_s.txt;

m = v9_s;

load v10_s.txt;

n = v10_s;
```

```
load v11_s.txt;
o = v11_s;
a1=[];
[ro1,cl1]=size(a);
for i=1:ro1
    for j=1:cl1
         a1=[a1;a(i,j)];
    end
end
b1=[];
for i=1:ro1
    for j=1:cl1
         b1=[b1;b(i,j)];
    end
end
c1=[];
for i=1:ro1
    for j=1:cl1
         c1=[c1;c(i,j)];
    end
end

d1=[];
for i=1:ro1
    for j=1:cl1
         d1=[d1;d(i,j);
    end
```

```
end
e1=[];
for i=1:ro1
     for j=1:cl1
         e1=[e1;e(i,j)];
     end
end
f1=[];
for i=1:ro1
    for j=1:cl1
        f1=[f1;f(i,j)];
    end
end
g1=[];
for i=1:ro1
    for j=1:cl1
        g1=[g1;g(i,j)];
    end
end
h1=[];
for i=1:ro1
    for j=1:cl1
        h1=[h1;h(i,j)];
    end
end
m1=[];
for i=1:ro1
```

```
   for j=1:cl1
        m1=[m1;m(i,j)];
   end
end
n1=[];
for i=1:ro1
    for j=1:cl1
        n1=[n1;n(i,j)];
    end
end
o1=[];
for i=1:ro1
    for j=1:cl1
        o1=[o1;o(i,j)];
    end
end

le=[a1 b1 c1 d1 e1 f1 g1 h1 m1 n1 o1];
t=eye(11);
net=newff(minmax(le),[50,11],{'tansig','purelin'},'traingd');
net.trainParam.epochs = 1000;
net.trainParam.goal=1e-2;
net.trainParam.lr=0.01;
net.trainParam.mc=0.1
net = train(net,le,t);
Y = sim(net,le);
```

```
cor=0;
for i=1:11
   index=1;
   for j=1:11
       if Y(j,i)>Y(index,i)
           index=i;
       end
   end
   if index==i
       cor=cor+1;
   end
end
'Train Cases Recognized'
cor
'Train Cases Recognition Accuracy'
cor/10*100

% Test Image

 load v1_s.txt;
a = v1_s;
a1=[];
for i=1:16
   for j=1:24
       a1=[a1;a(i,j)];
   end
end
```

```
 a1
 match=sim(net,a1);

'test image output'
'Output vector for test image'
 match
 index=1;
 for j=1:11
    if match(j)> match(index)
        index=j;
    end
 end
 index
 [a,b]=size(match);
 f = fopen('ds3.txt','w');
 for i = 1 : a
    for j = 1 : b
        fprintf(f,'%f ', match(i,j));
    end
    fprintf(f,'\r\n');
 end
 fclose(f);
```