

## Sessió 12

### Xavier Martín Ballesteros i Adrià Cabeza Sant'Anna

Primerament, hem utilitzat un sol HOG com a model del 0 i un sol HOG com a model de l'1. Com que ens donava poca precisió, hem comprovat quins casos eren els que fallava més.

En el cas del 0, hem agafat també com a model un 0 rotat una mica. En el cas de l'1, hem agafat també com a model un 1 rotat una mica i amb base.

Amb aquests 2 models per cada una de les xifres hem començat a classificar les imatges de testing. Per cada imatge de testing, hem calculat el seu HOG i hem determinat amb quin model tenia la mínima distància (cityblock).

Si només provem imatges amb nombres digitals, obtenim un 100% de precisió. El 86% de precisió l'hem obtingut utilitzant imatges a mà tant pel 0 com per l'1.

```
bins = 16;

model_00 = getHog("Digit 0.jpg", bins);
model_01 = getHog("esun0_2.tif", bins);
% bar(model_00)
% bar(model_01)

model_10 = getHog("Digit 1.jpg", bins);
model_11 = getHog("esun1_7.tif", bins);
% bar(model_10)
% bar(model_11)

test0 = ["esun0_1.tif", "esun0_2.tif", "esun0_3.tif", "esun0_4.tif", "esun0_5.tif", ...
         "esun0_6.tif", "esun0_7.tif", "esun0_8.tif", "esun0_10.tif", "esun0_19.tif", ...
         "esun0_9.tif"];

test1 = ["esun1_1.tif", "esun1_2.tif", "esun1_3.tif", "esun1_4.tif", "esun1_5.tif", ...
         "esun1_6.tif", "esun1_7.tif", "esun1_8.tif", "esun1_16.tif", "esun1_15.tif", ...
         "esun1_9.tif"];

correctes0 = 0;
correctes1 = 0;
[row0 col0] = size(test0);

for i = 1:col0
    hog = getHog(test0(i), bins);
    % figure, bar(hog)
    dist00 = pdist2(model_00', hog', 'cityblock');
    dist01 = pdist2(model_01', hog', 'cityblock');
    dist10 = pdist2(model_10', hog', 'cityblock');
    dist11 = pdist2(model_11', hog', 'cityblock');

    dist0 = min(dist00, dist01);
    dist1 = min(dist10, dist11);

    if (dist0 < dist1)
```

```

        correctes0 = correctes0 + 1;
    end
end

[row1 col1] = size(test1);
for i = 1:col1
    hog = getHog(test1(i), bins);
    % figure, bar(hog)
    dist00 = pdist2(model_00', hog', 'cityblock');
    dist01 = pdist2(model_01', hog', 'cityblock');
    dist10 = pdist2(model_10', hog', 'cityblock');
    dist11 = pdist2(model_11', hog', 'cityblock');

    dist0 = min(dist00, dist01);
    dist1 = min(dist10, dist11);

    if (dist0 > dist1)
        correctes1 = correctes1 + 1;
    end
end

percentatge = (correctes0 + correctes1)/(col0 + col1) * 100;
disp("Percentatge d'encert: " + num2str(percentatge) + '%')

```

Percentatge d'encert: 86.3636%

```

function [hist] = getHog(im, bins)
    im_test = imread(im);

    sob = fspecial("sobel");
    soby = sob/4;
    sobx = soby';

    grady = imfilter(double(im_test), soby, 'conv');
    gradx = imfilter(double(im_test), sobx, 'conv');

    arg = atan2(grady, gradx);

    neg = (arg < 0);
    arg(neg) = arg(neg) + 2*pi;

    angle = uint8((arg+pi)/2/pi*254);
    angle = angle/bins;

    mod = sqrt(gradx.^2+grady.^2);
    mask = (mod==0);
    angle(mask) = 255;
    h = imhist(angle);
    % El while és innecessari. No modifica cap valor de l'histograma.
    % L'utilitzem perquè quedi un histograma més bonic (no apareguin els 0s
    % inicials).

```

```
j = 1;
while h(j) == 0
    j = j + 1;
end

hog = h(j:end/bins-1);
hog = hog/sum(hog);
hist = hog;
end
```