**Project : Face recognition**

**Histogram of oriented gradients algorithm**

**Step 1: Preprocessing,**

We need to preprocess the image and bring down the width to height ratio to 1:2.

**Step 2: Calculating Gradients (direction x and y),**

Gradients are the small change in the x and y directions.



To find gradients of above matrix, subtract left pixel value to the right pixel value (gradient in x direction) and subtract upper pixel value to lower pixel value (gradient in y direction).

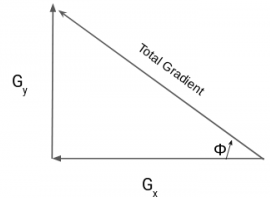
Hence the resultant gradients in the x and y direction for this pixel are:

Change in X direction (Gx) = 89 – 78 = 11

Change in Y direction (Gy) = 68 – 56 = 8

**Step 3: Calculate the Magnitude and Orientation**

**To find magnitude** we use Pythagoras theorem



For the previous example, we had Gx and Gy as 11 and 8. Let’s apply the Pythagoras theorem to calculate the total gradient magnitude:

Total Gradient Magnitude = √[(Gx)2+(Gy)2]

Total Gradient Magnitude = √[(11)2+(8)2] = 13.6

**Calculate direction**

tan(Φ) = Gy / Gx

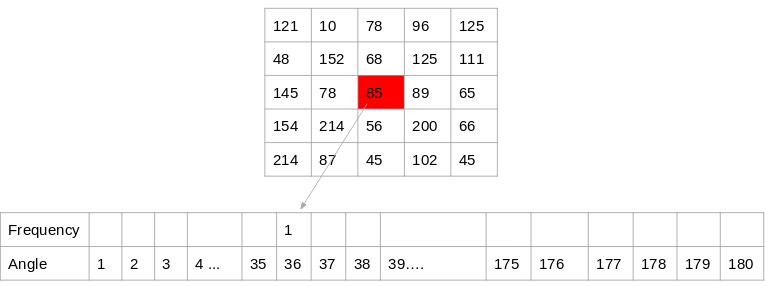
Hence, the value of the angle would be:

Φ = atan(Gy / Gx)

The orientation comes out to be 36 when we plug in the values

**Step 4: Calculate Histogram of Gradients in 8×8 cells**

We get the features (or histogram) for the smaller patches which in turn represent the whole image.



**Step 5: Normalize gradients in 16×16 cell**

Because some portion of the image would be very bright as compared to the other portions.

**Matlab Code**

I1 = imread('gantrycrane.png');

[hog1,visualization] = extractHOGFeatures(I1,'CellSize',[32 32]);

subplot(1,2,1);

imshow(I1);

subplot(1,2,2);

plot(visualization);

