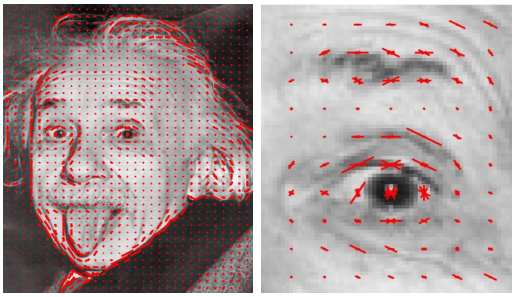
(a) Input Image (b) Gradient Magnitude (c) Gradient (d) Zoomed eye (e) Zoomed Tongue

Figure 1: Visualization of (a) Input Image (b) Gradient Magnitude (c)-(e)Visualization of gradients for every 2nd pixel



(a) HOG on input Image (b) HOG on zoomed in eye

Figure 2: Histogram of oriented gradients. HOG feature is extracted and visualized for (a) Entire Image (b) Zoom in Image. The orientation and magnitude of the red lines represent the gradient components in a local cell

Summary-Homework:1

The problem is to take an input image and extract its HOG feature descriptor. Here we considered an Einstein image of dimensions $345 \times 264 \times 3$ (3 channels) and implemented the algorithm proposed by Dalal and Trigg[] to obtain the HOG.

To this end, I wrote the following functions in MATLAB: HOG.m, GetDifferentialFilter.m, GetGradient.m, BuildHistogram.m, GetBlockDescriptor.m which are compliant with the specifications mentioned in the problem sheet. First, the x-component of the gradient is obtained by convolving the grayscale image with 3×3 kernel i.e $[1, 0, -1; 1, 0, -1; 1, 0, -1]$ and the y-component is obtained by convolving with $[1, 1, 1; 0, 0, 0; -1, -1, -1]$. Since, we zero pad the image, the size of the gradient image remains same as input image size. It is to be noted that x-component of the gradient captures the vertical edges while the y-component captures the horizontal edges in an image. Post gradient calculation, a histogram of oriented gradients based on the angles of the gradient is built for every cell which in our case is a grid of 8x8 pixels. [0 180] interval has been discretized into 6 bins and therefore the output returned is a vector of size 43x33x6. This is HOG without normalization. GetBlockDescriptor.m normalizes the obtained gradients in the previous step by considering a block size of 2x2 cells and returns HOG, which is of size 42x32x24.
The visualizations of the Gradient have been done using quiver(). Fig.1(c) plots the gradient on the input image. The visualization has been done for every second pixel. As we can see, the gradient directions are perpendicular to the edge directions. The gradient can be very clearly seen plotted as horizontal lines on the tongue region. Fig.2(a) and Fig.2(b) gives the visualizations of the HOG on the input image.The vectors are plotted on the original image with the centre of each cell being the starting point and the direction being the mid-point of each bin angle, i.e. 0°for bin 1, 30°for bin 2 and so on. Here, the line directions are perpendicular to the gradient directions to show edge alignments. The magnitudes have been re-scaled for visualization purposes.