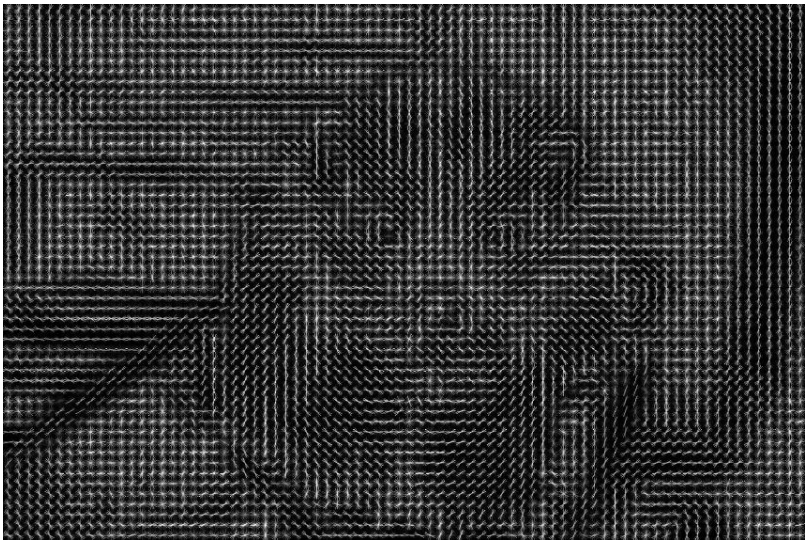
DIP Assignment 5

Task 1: Histogram of Gradient Feature descriptors.
[Code used to generate all output present in file **hog.m]**

HOG decomposes an image into small squared cells, computes an histogram of oriented gradients in each cell, normalizes the result using a block-wise pattern, and return a descriptor for each cell. The main parameters used in HOG are the cell size(region around a pixel) and the number of orientations to compute the histogram. For a given image HOG returns an image of cells, where the distribution of gradients in different directions is shown:

Original Image

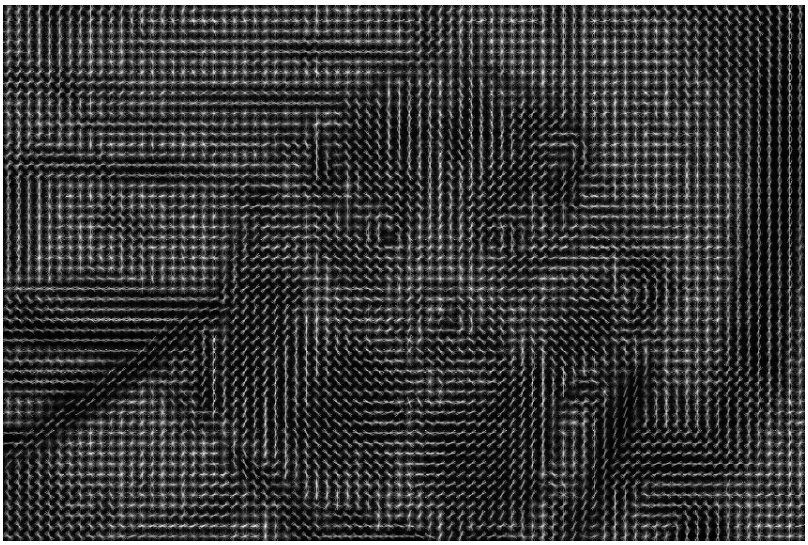
HOG result(cell_size=8, orientations=8)



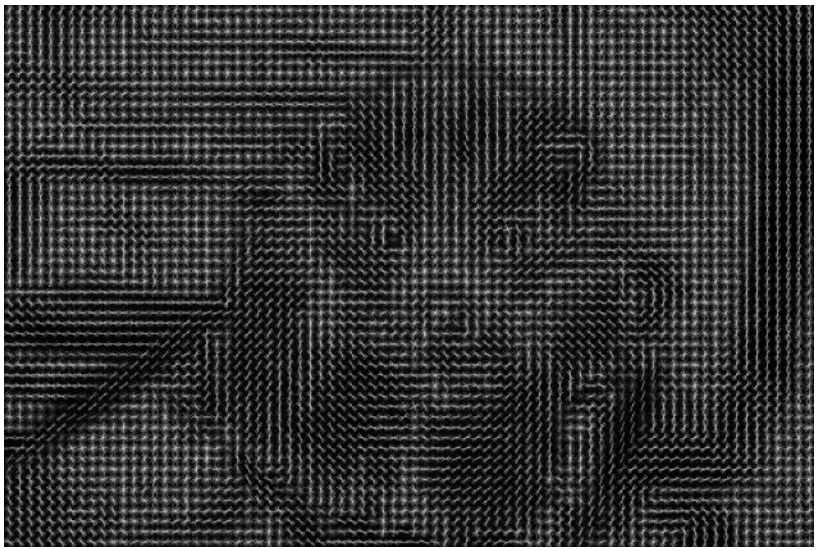
NOTE: All images have been resized to fit in report, original images can be generated using the scripts provided.

There are two variants of HOG present in VLFeat, they are the UoCTTI variant (used by default) and the original Dalal-Triggs variant. Both variants give similar results with small differences.

UoCTTI variant

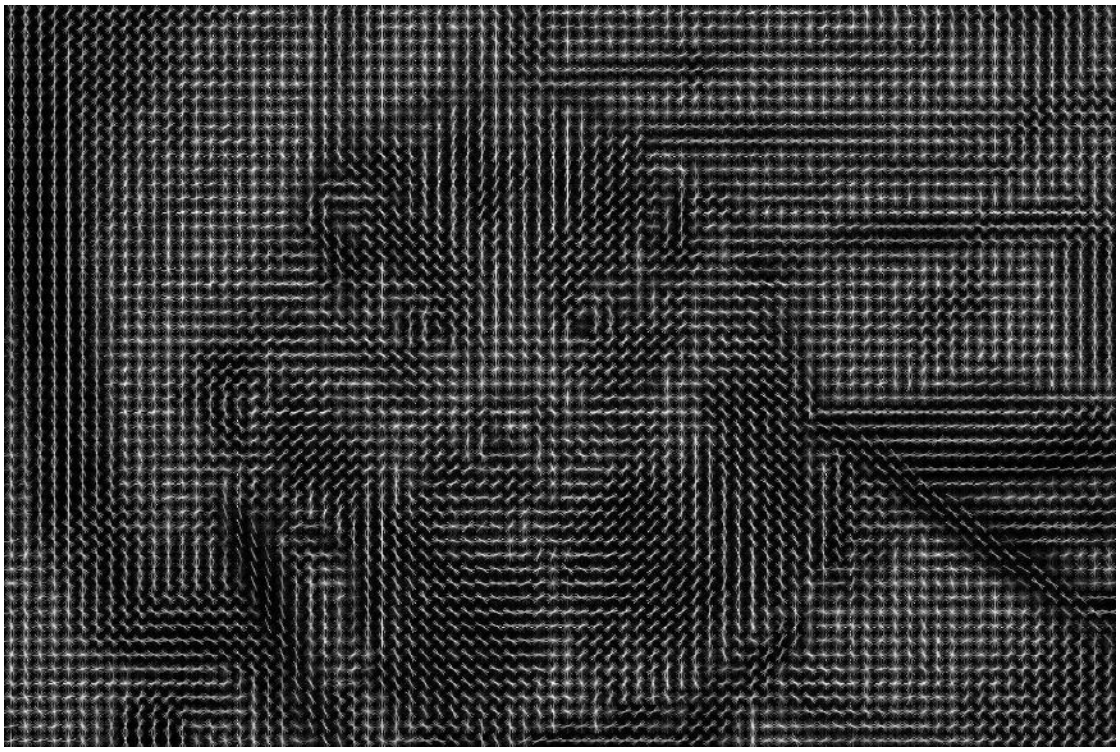


Dalal-Triggs variant

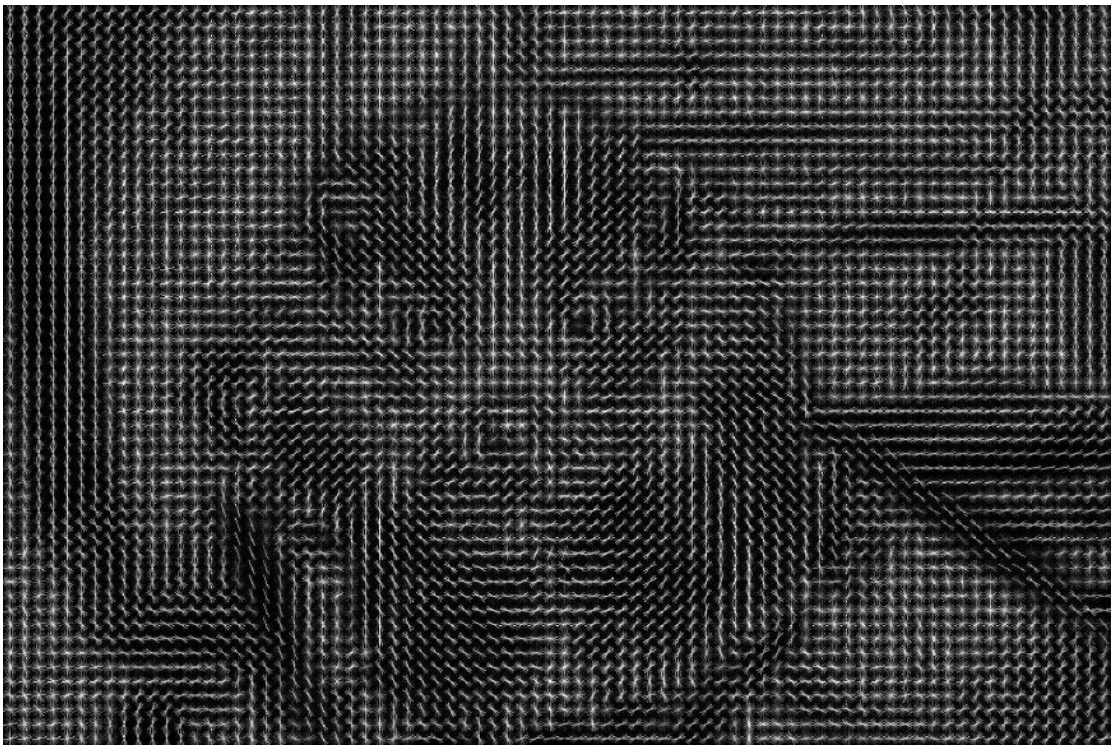


Flipping the image has no effect on HOG. HOG of flipped image is same as the flipped output of HOG of original image.

HOG of flipped image

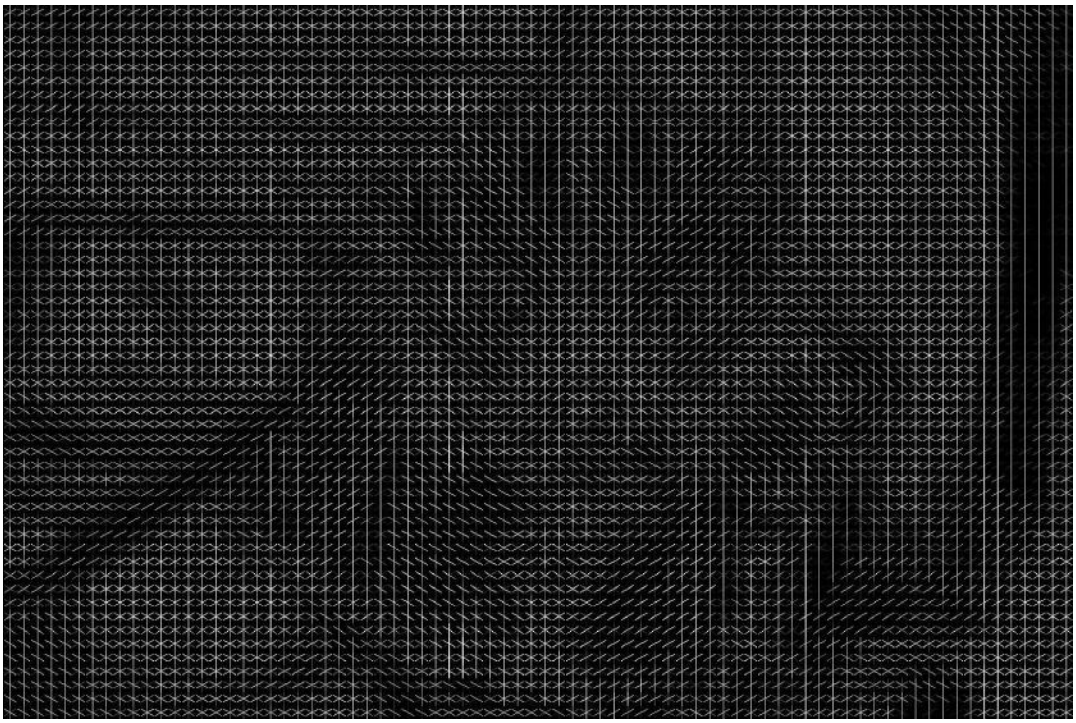


Flipped result of HOG image.

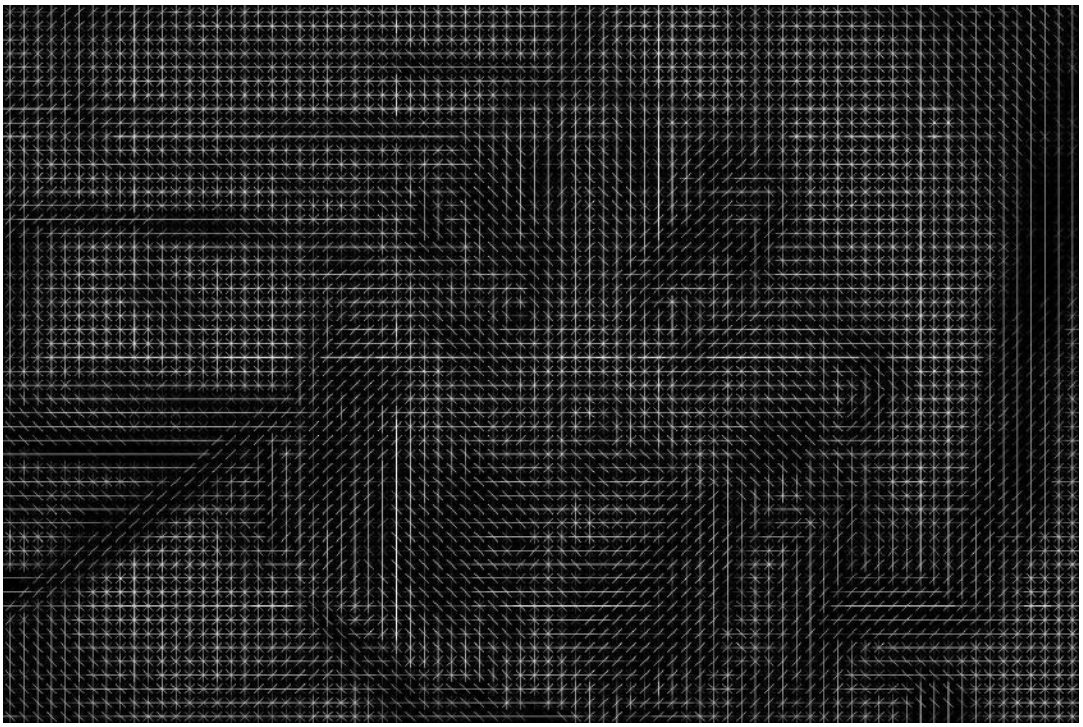


Varying the cell size in hog gives the same distribution computed over larger cells. The smaller the cell size the better the more descriptive the feature. However varying the number of orientations considered gives a much more descriptive output for an image. Here are the same image with a fixed cell size and varying number of orientations.

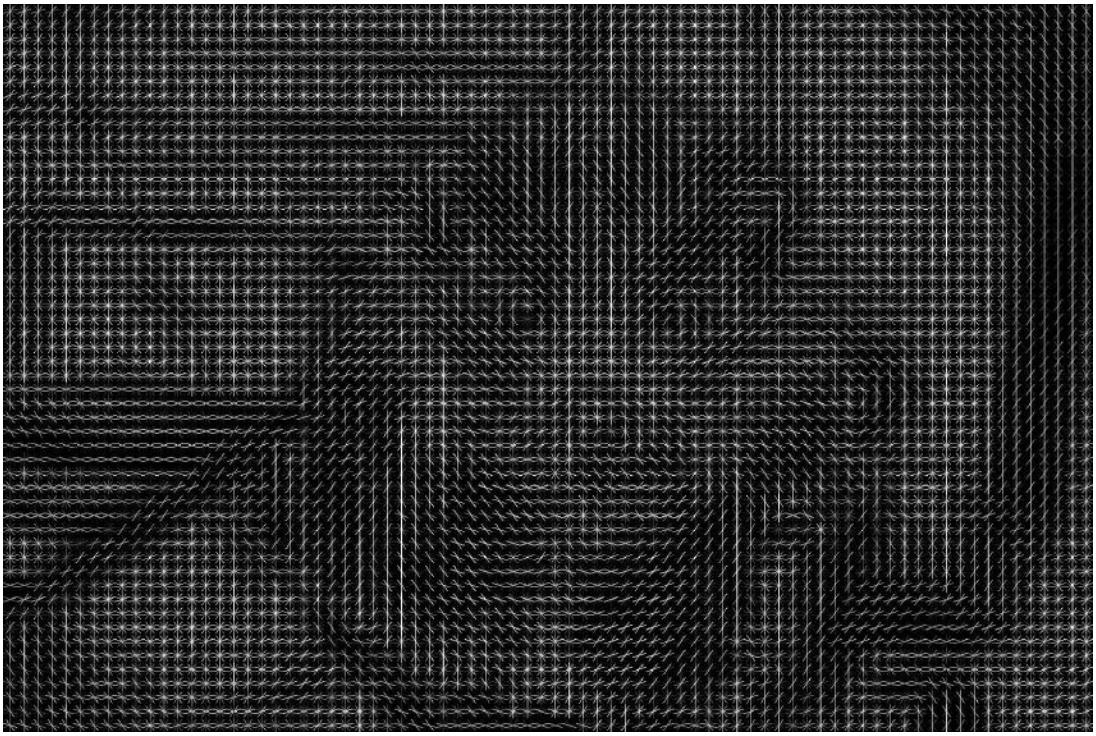
3 directions



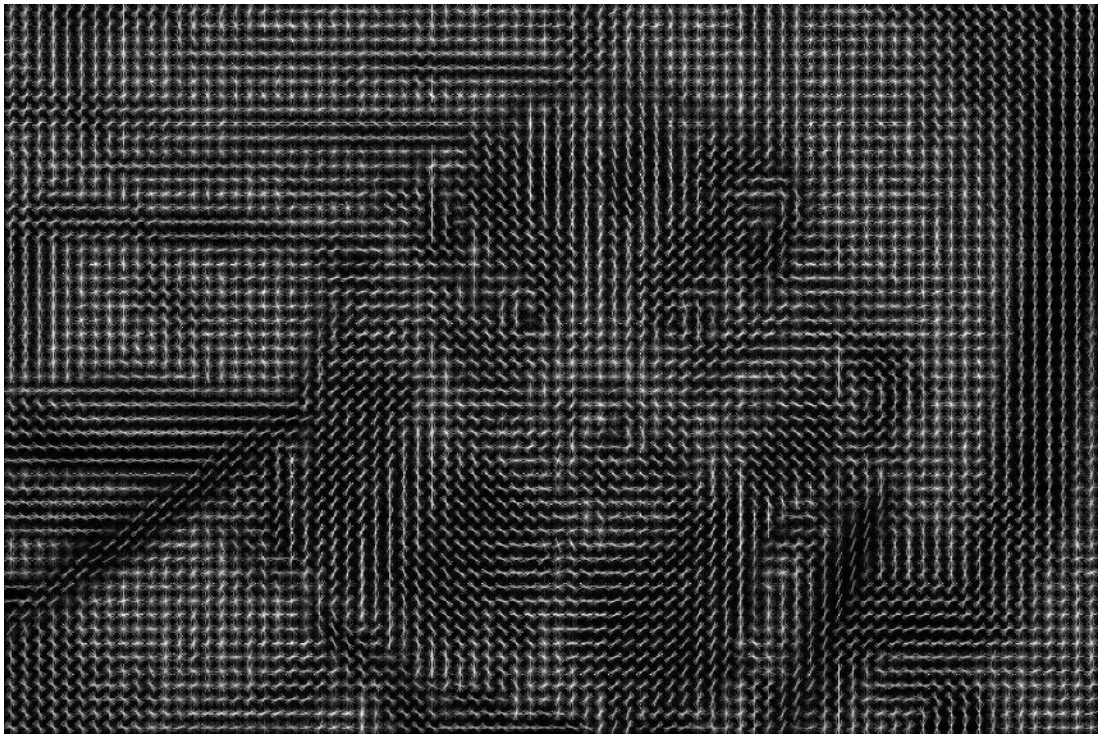
4 directions



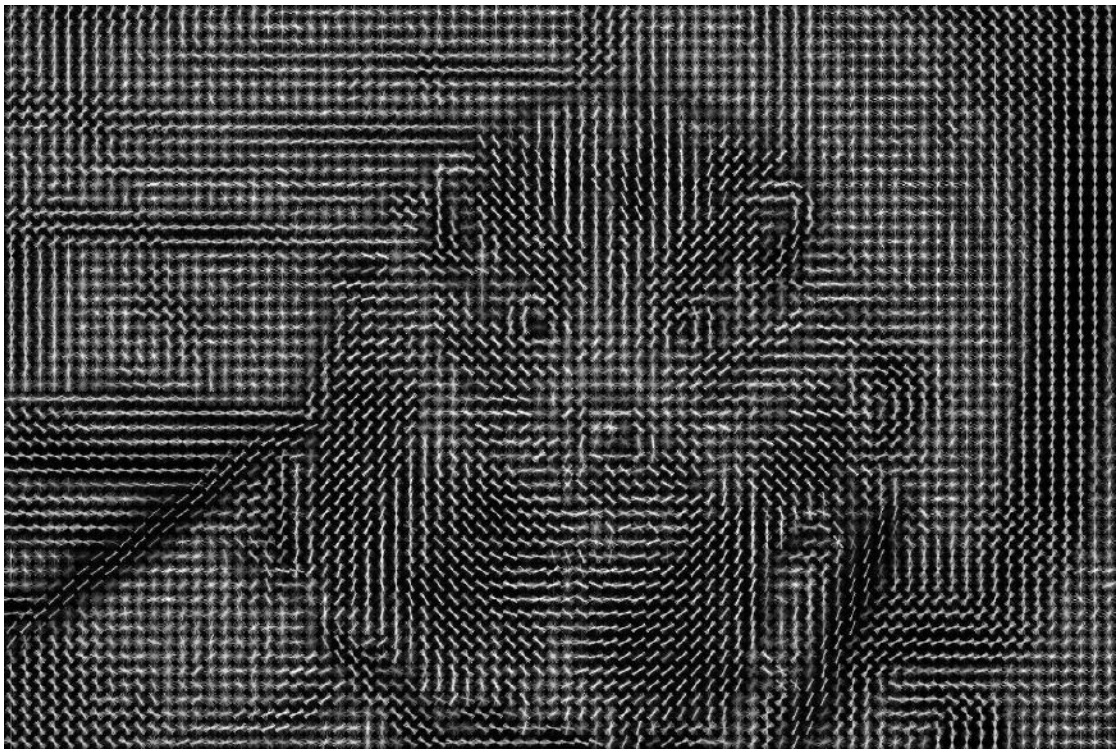
5 direction



9 directions

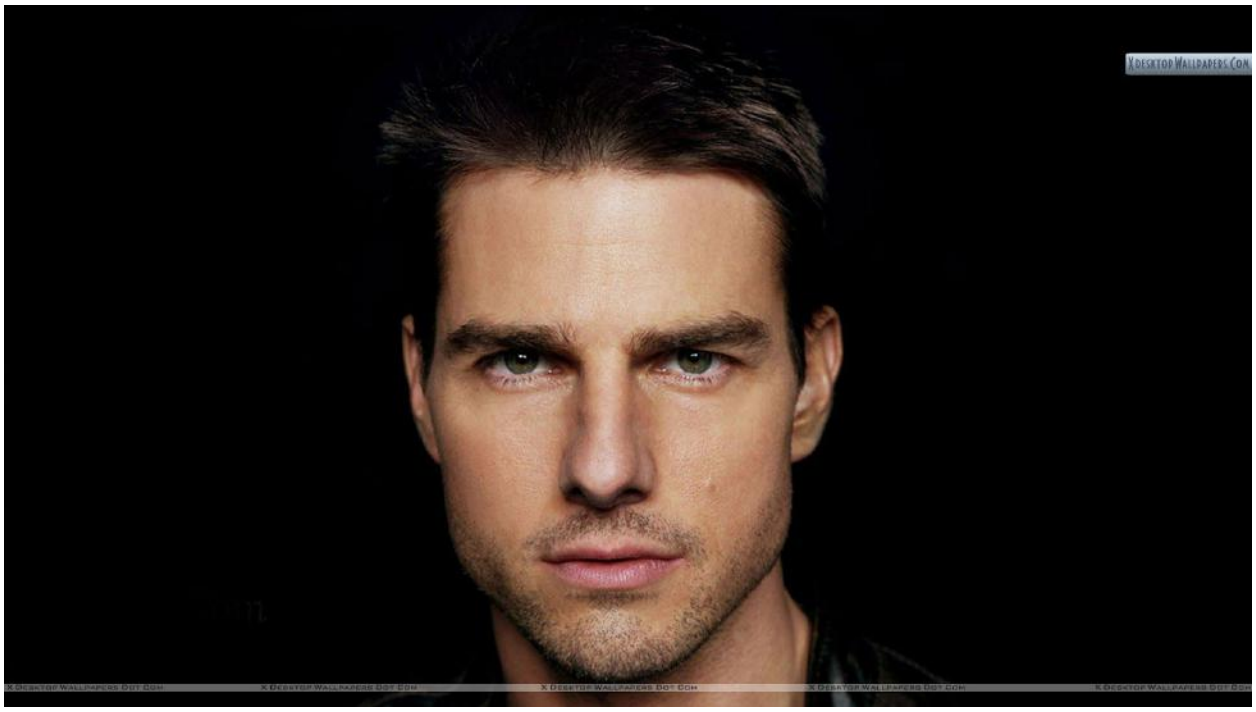


21 directions



The same experiments were done for 2 other images:

Original Image

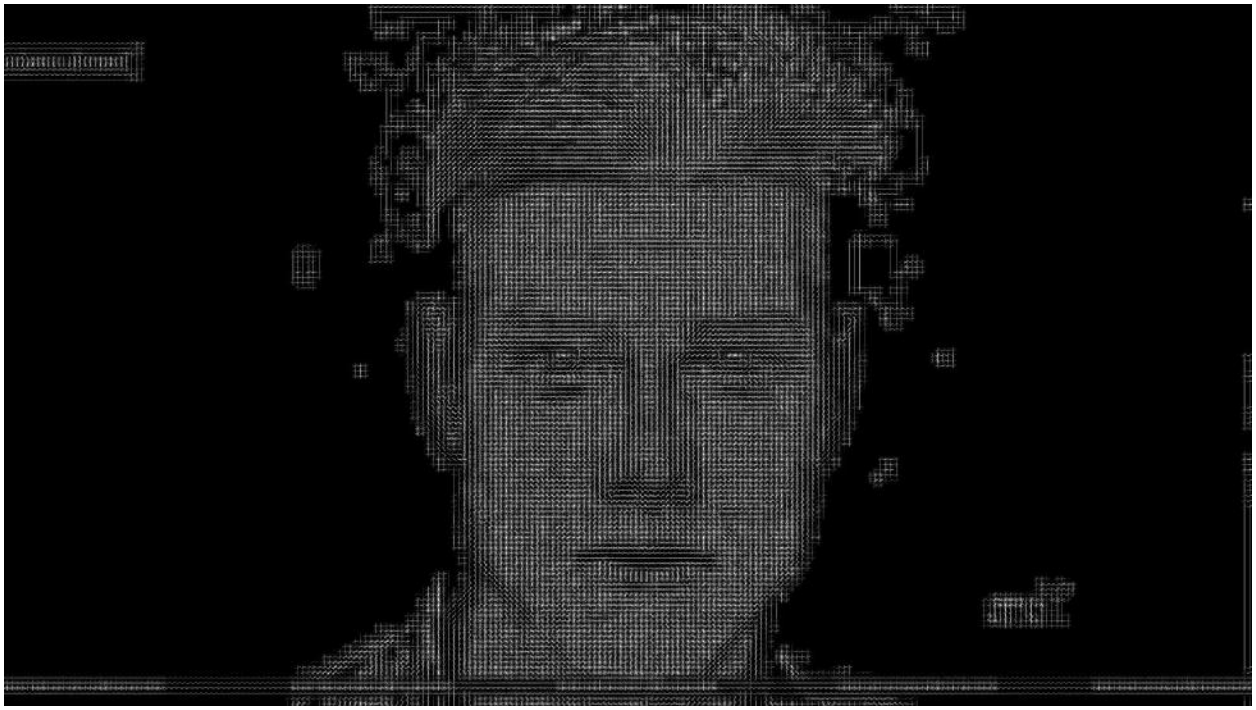


UoCTTI HOG(cellsize=8, orientations=8)

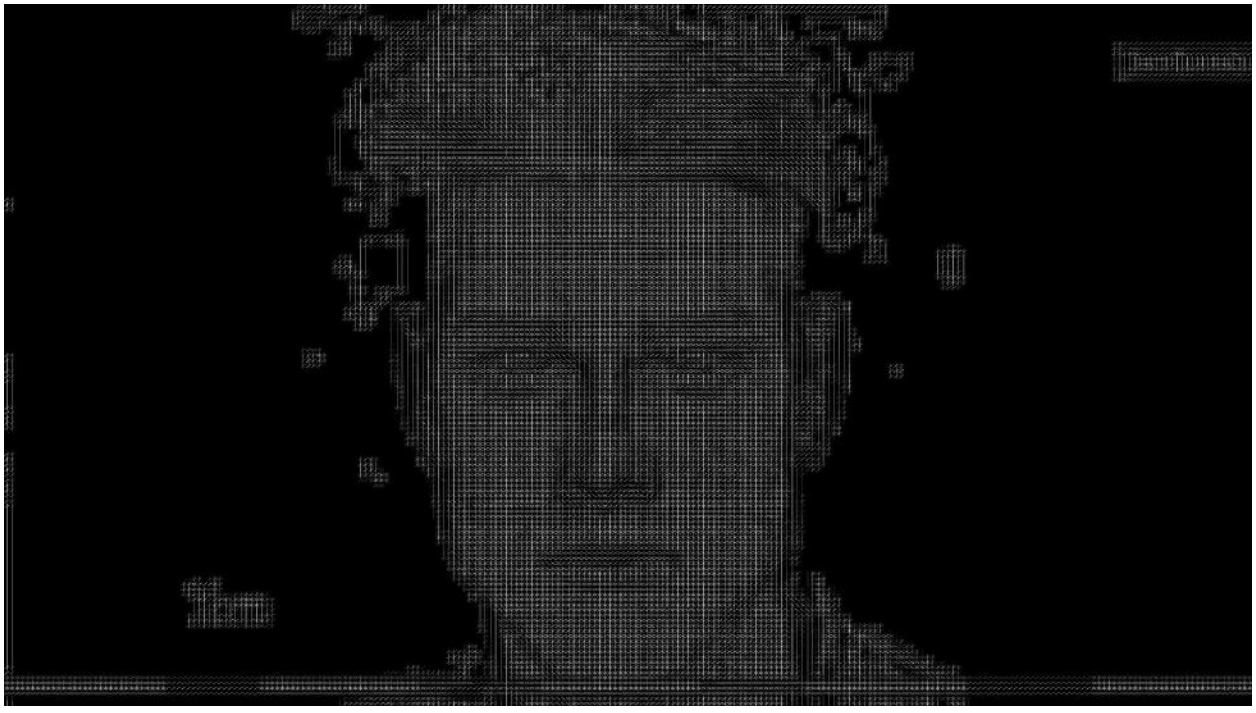
Dalal Triggs HOG(cellsize=8, orientations=8)

HOG of flipped image.

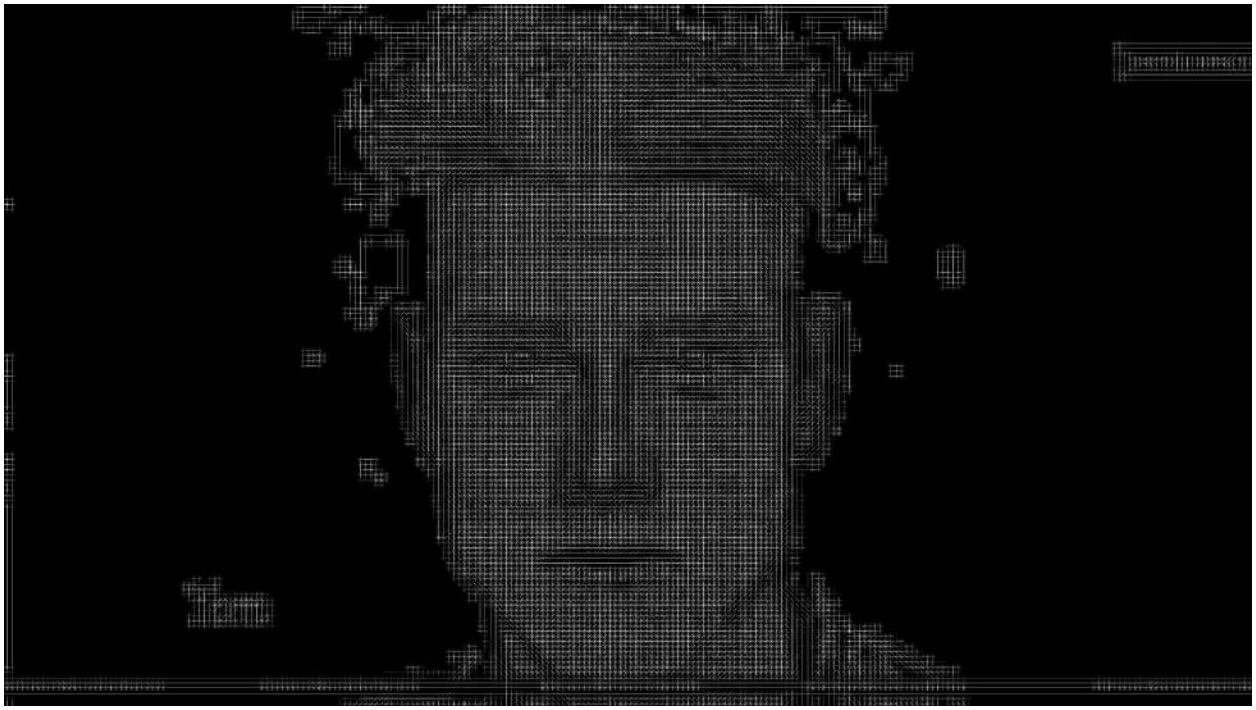
Flipped image of original HOG



HOG(orientations=3)



HOG(orientations=4)



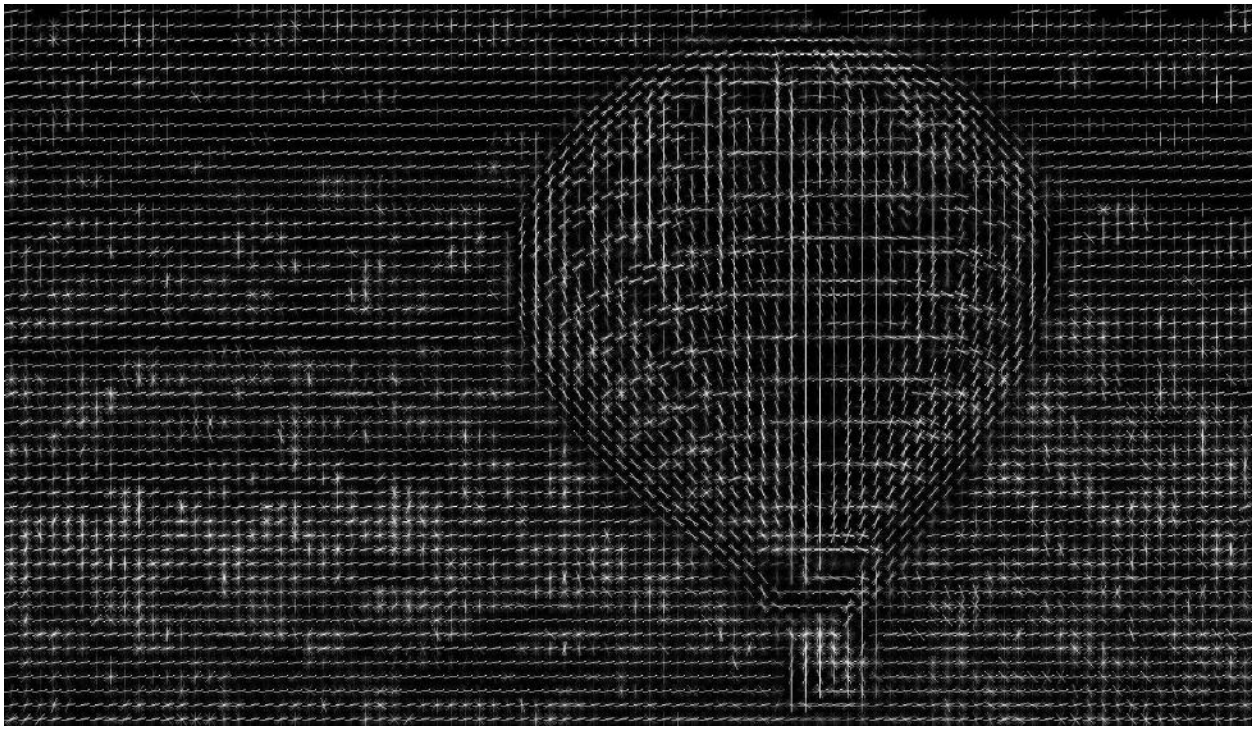
HOG(orientations=5)

HOG(orientations=9)

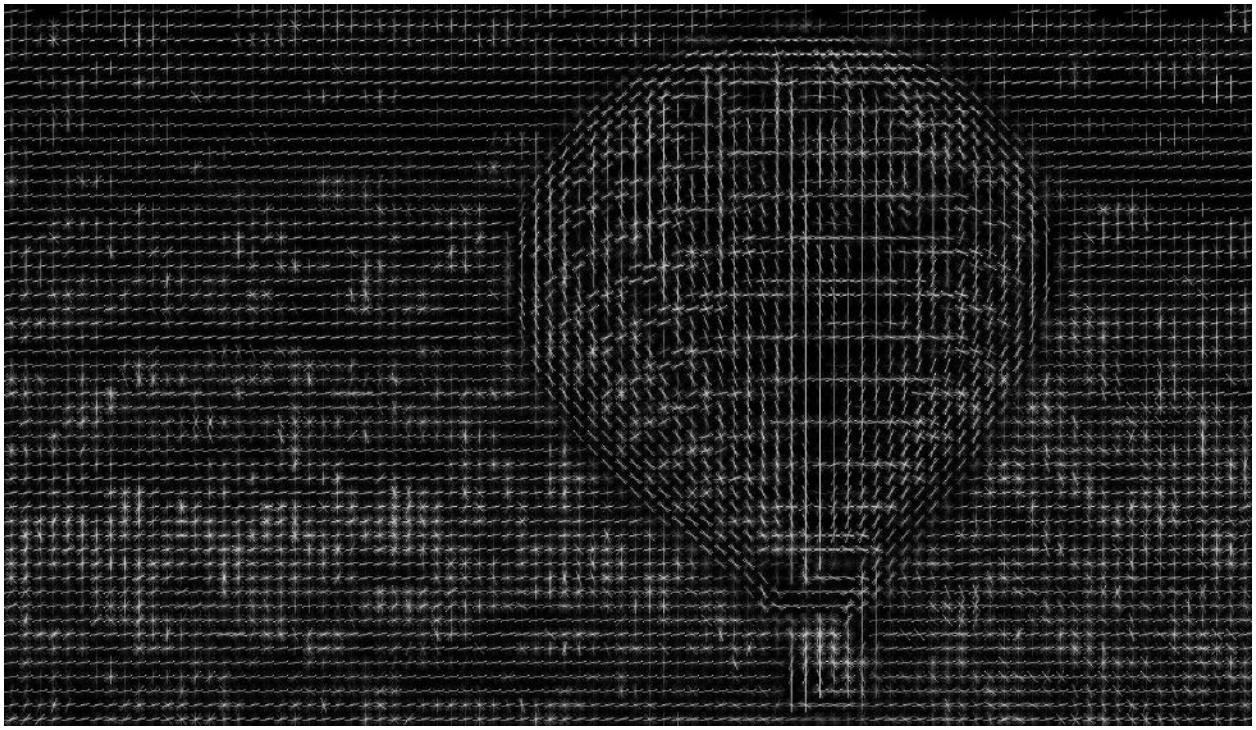
HOG(orientations=21)

Original Image

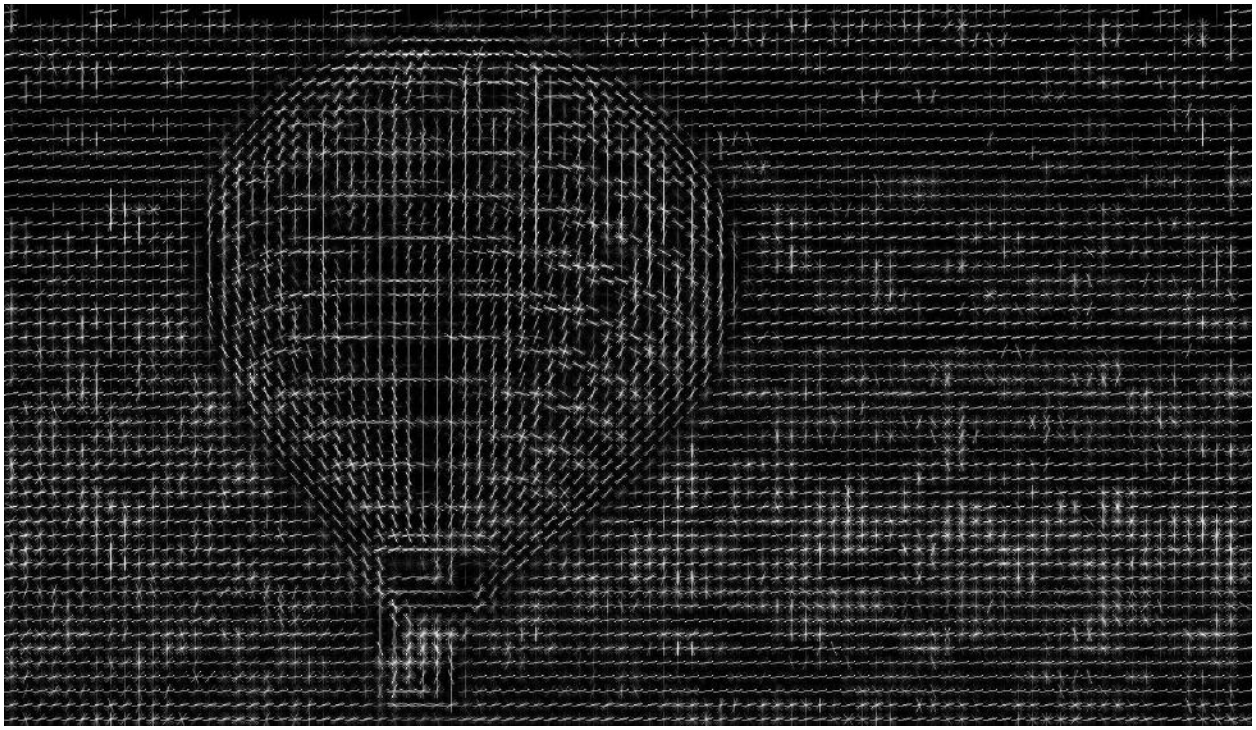
UoCTTI HOG(cellsize=8, orientation=8)



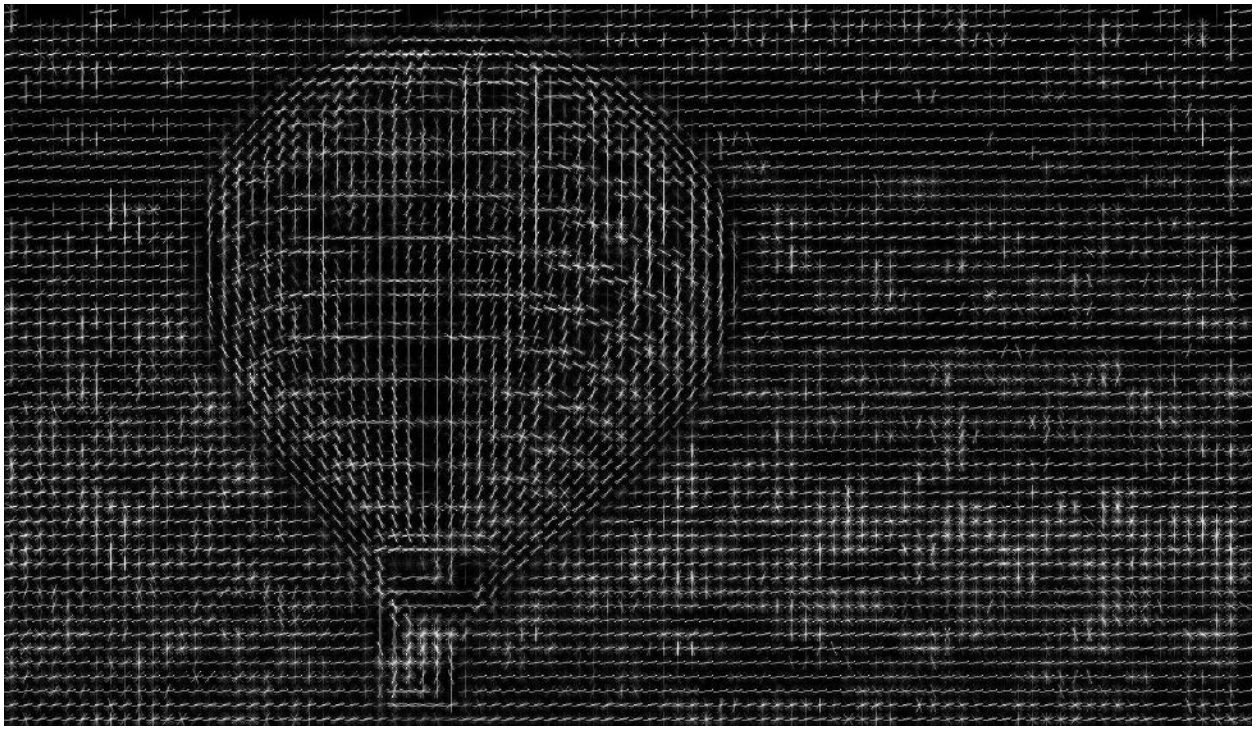
Dalal Triggs HOG(cellsize=8, orientations=8)



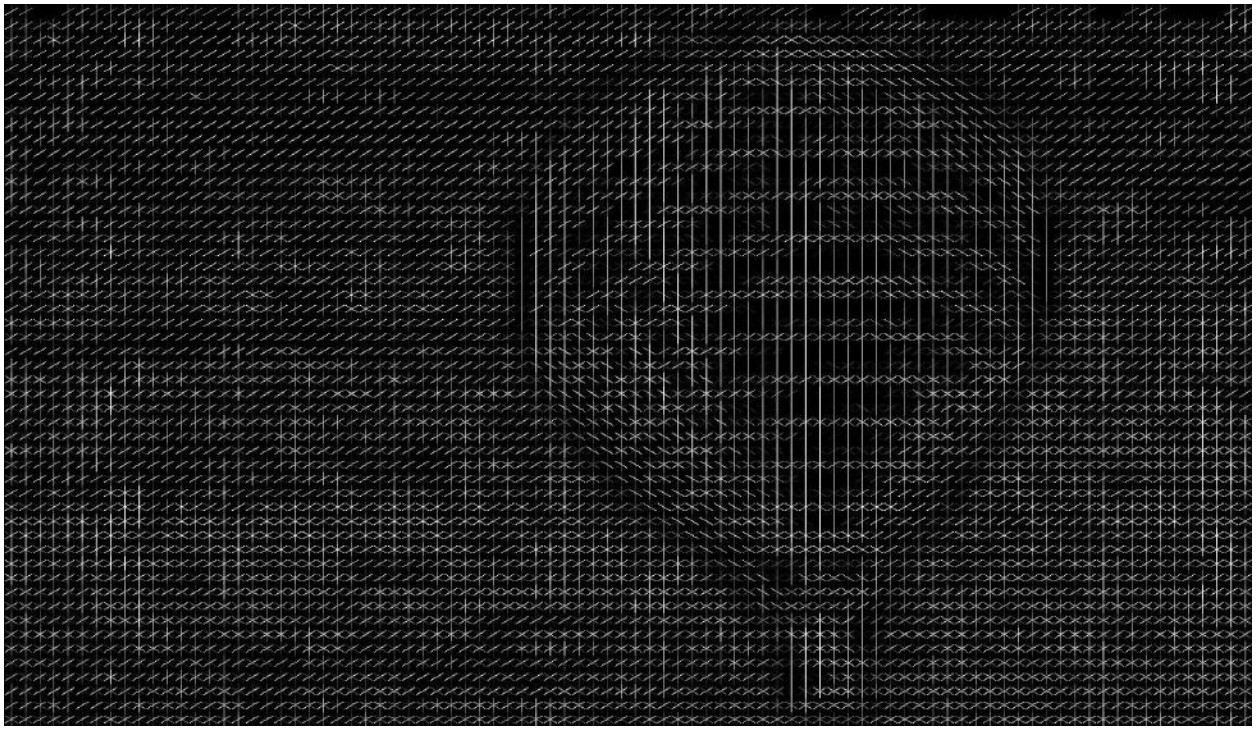
HOG of flipped Image



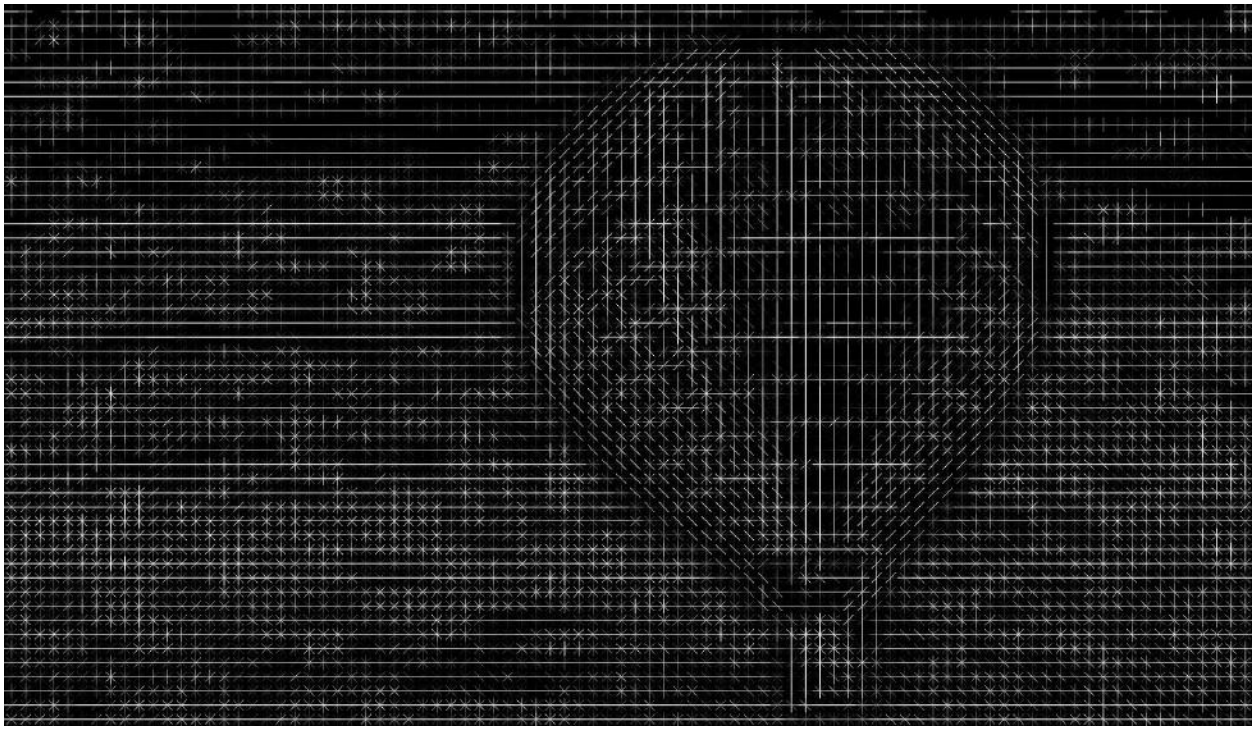
Flipped HOG of original image



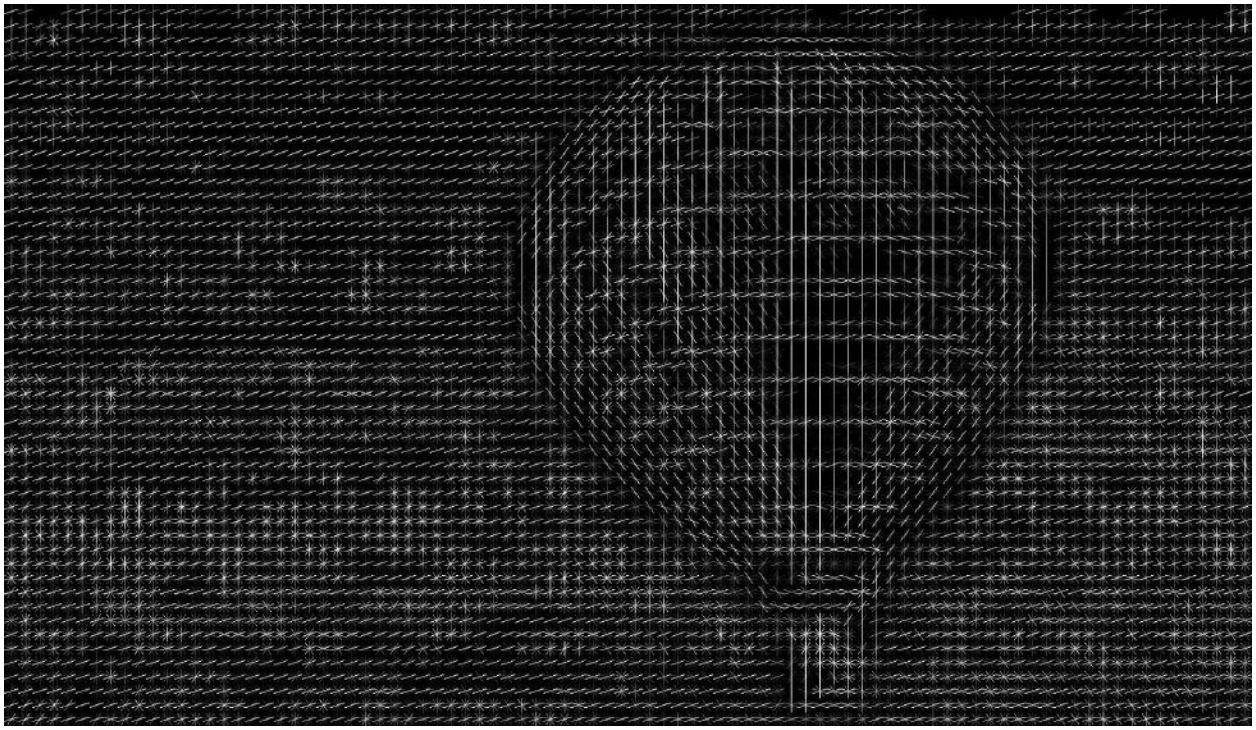
HOG(orientations=3)



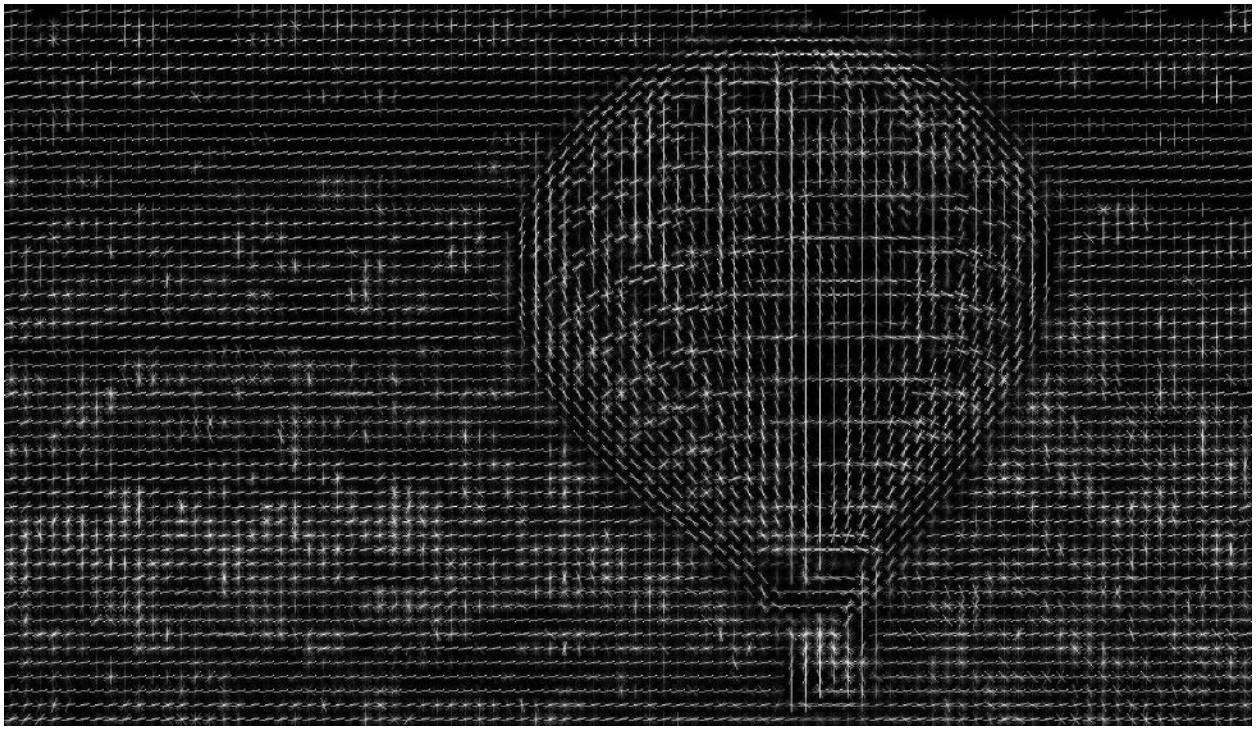
HOG(orientations=4)



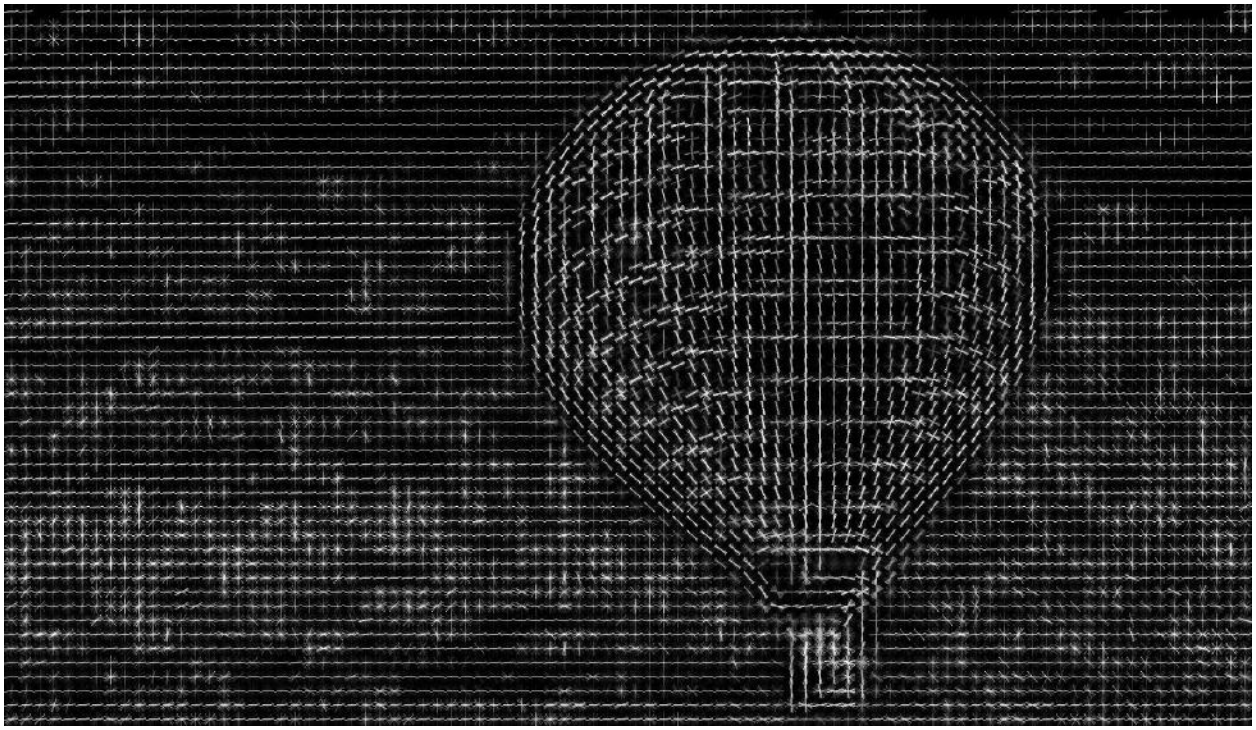
HOG(orientations=5)



HOG(orientations=9)



HOG(orientations=21)

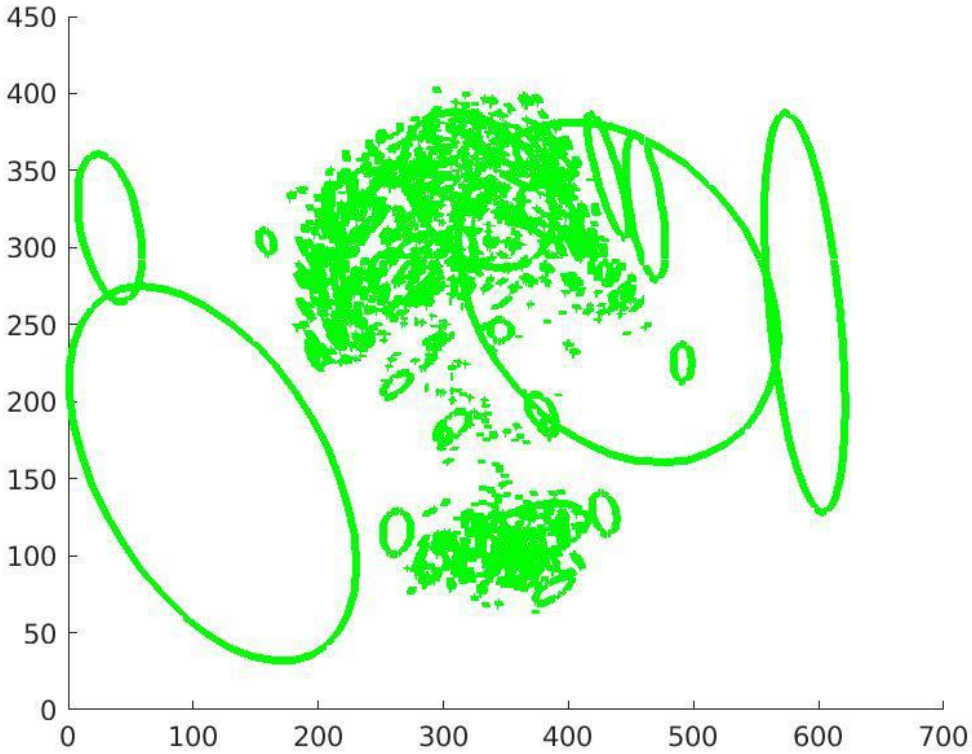


Task 2: Maximally Stable Extremal Regions (MSER)
[Code used to generate all output present in file **mser.m]**

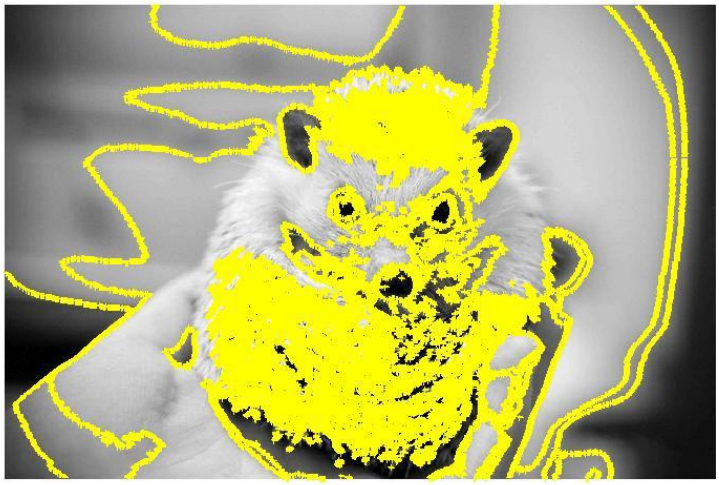
MSER algorithm extracts a number of co-variant regions, called MSERs. An MSER is a stable connected component of some level sets of the image. Elliptical frames are attached to the MSERs by fitting ellipses to the regions.

Original Image

Elliptical frames by region



Plotted MSER regions



MSERs are extracted for both dark-on-bright regions and bright-on-dark regions. The default image is dark-on-bright. The bright-on-dark image is shown below:

The value of delta in MSER determines the stability of a region. As we increase delta, fewer and fewer regions are detected.

MSER(delta=32)

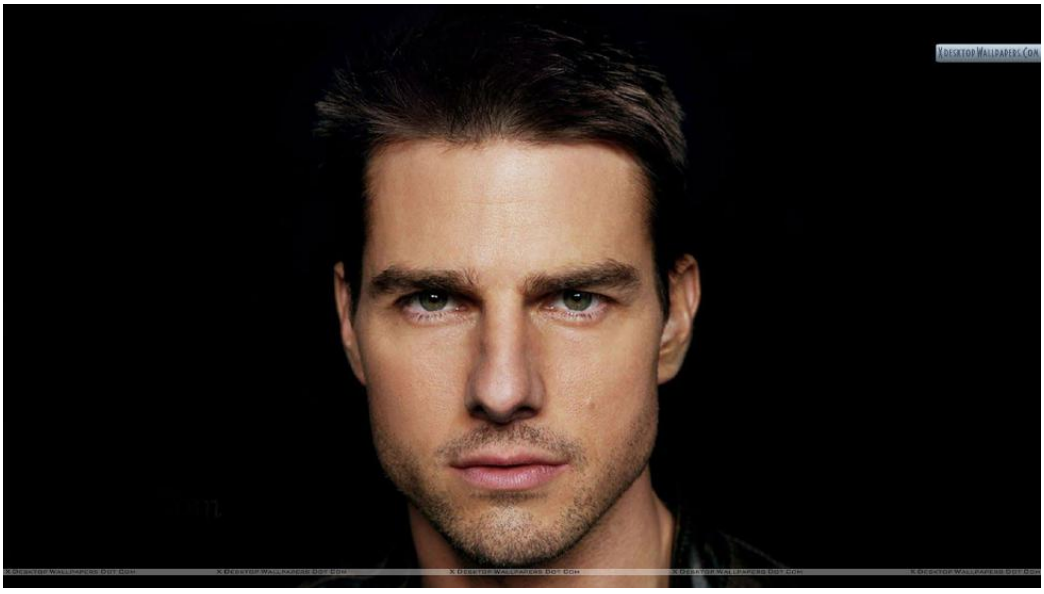
MSER(delta=64)

MSER(delta=96)

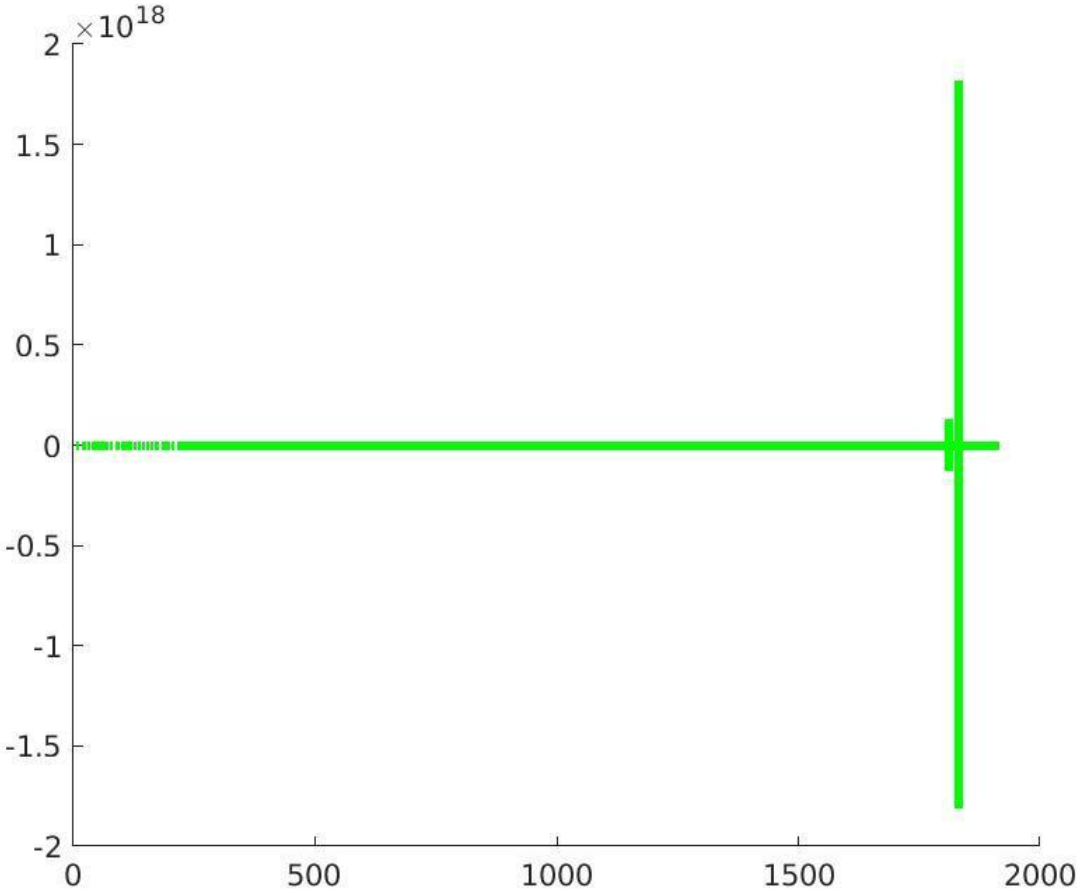
No regions detected after 96.

The experiment was repeated for two other images.

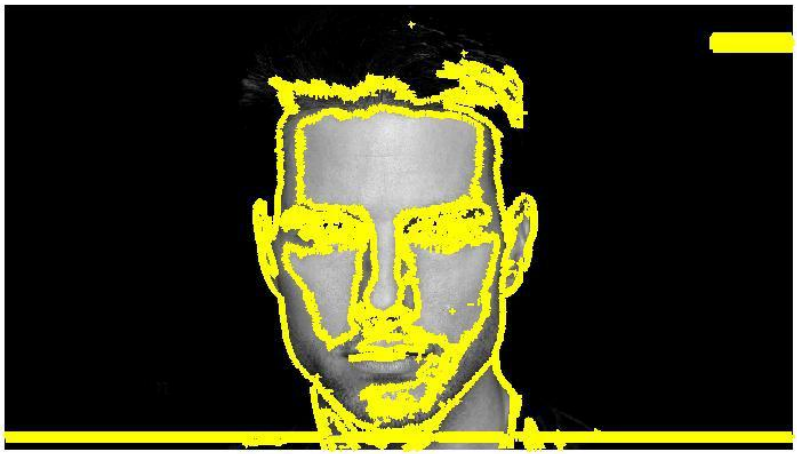
Original Image



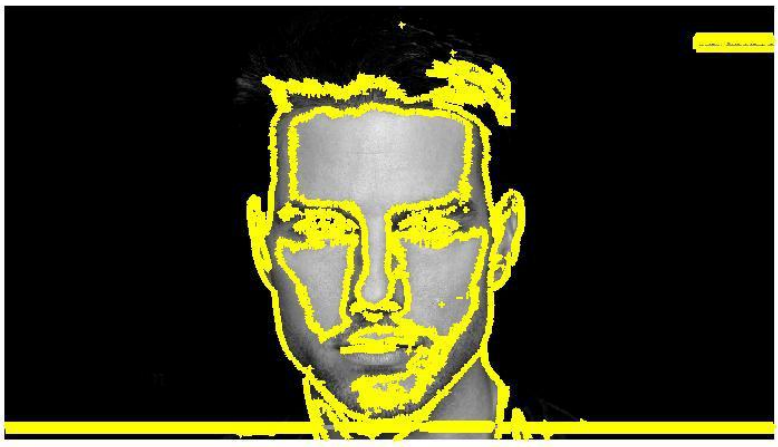
Elliptical frames by region



MSER(dark-on-bright, delta=10)

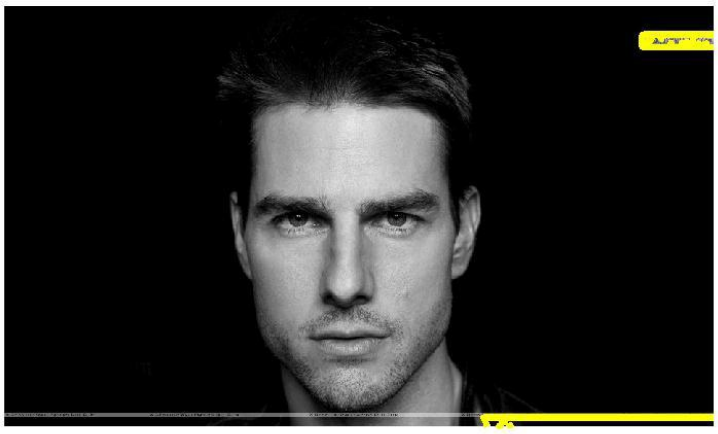


MSER(bright-on-dark, delta=10)

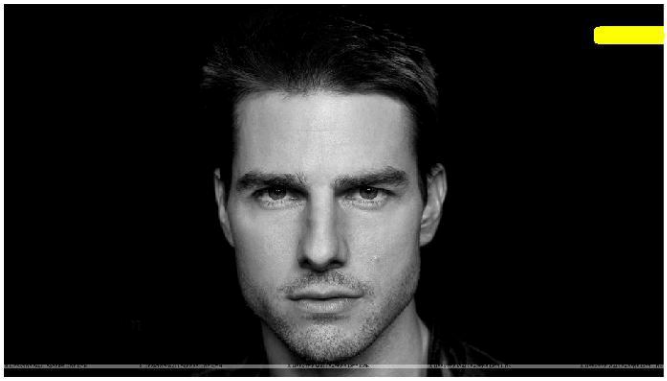


MSER(delta=32)

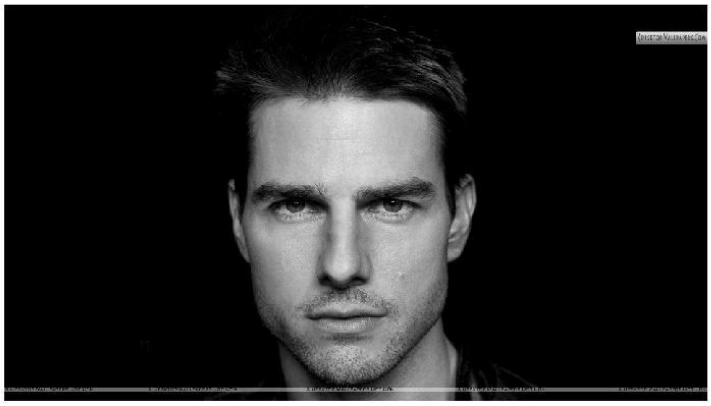
MSER(delta=64)



MSER(delta=96)



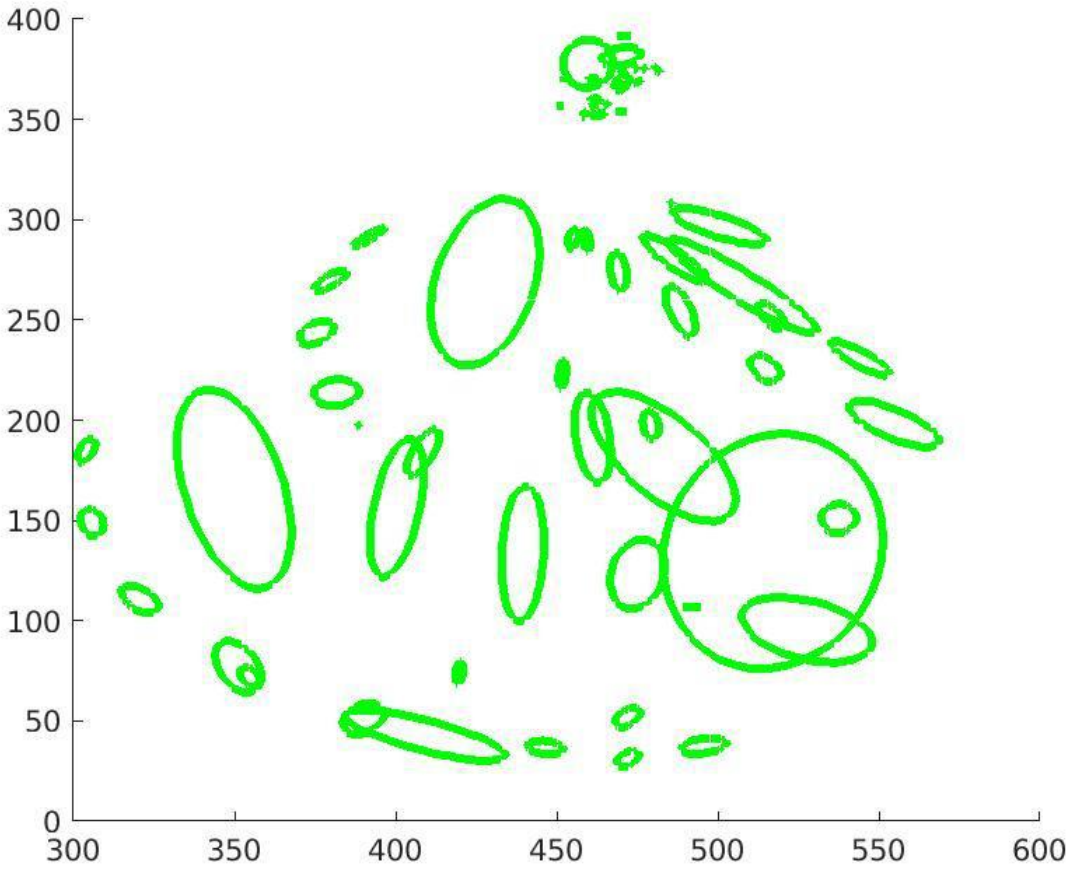
MSER(delta=160)



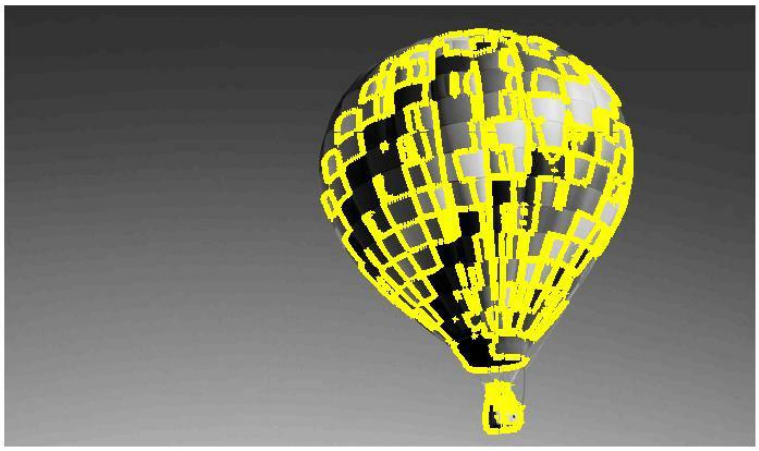
No regions detected after delta=160.

Original Image

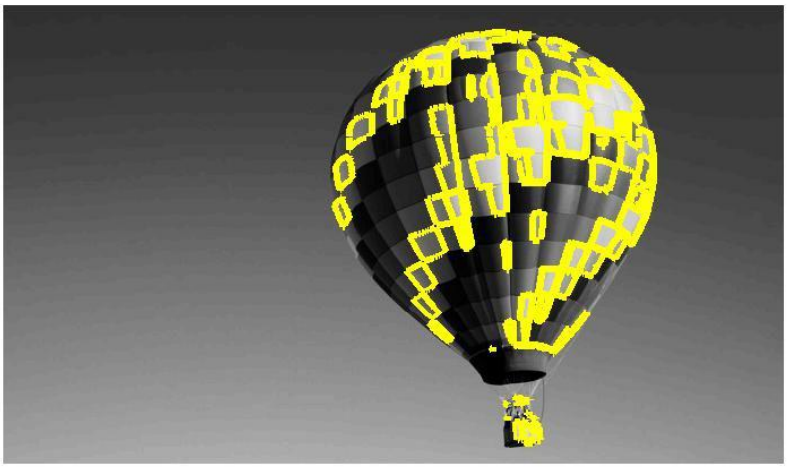
Elliptical frames by region



MSER(dark-on-bright, delta=10)



MSER(bright-on-dark, delta=10)



MSER(delta=32)

MSER(delta=64)

MSER(delta=96)

No stable regions after delta=96.