Image Stitching for Panorama View

By:

Arpana Alka

Indian Institute of Space Science and Technology

Guide:

Dr Rama Rao (IIST)

Introduction

- Image Stitching is basically combining two or more different images to form a single panorama view.
- Panorama is derived from a Greek word, Pan=Everything and Horama=to view.
- Camera has a field view of 50X35 degrees, whereas human eye has 200X135 degrees.

Simple Image Stitching Code

a= imread('left.jpg'); %read the left part of the image

b=imread('right.jpg'); %read the right part of the image

sa= size(a) %get the size of the left image

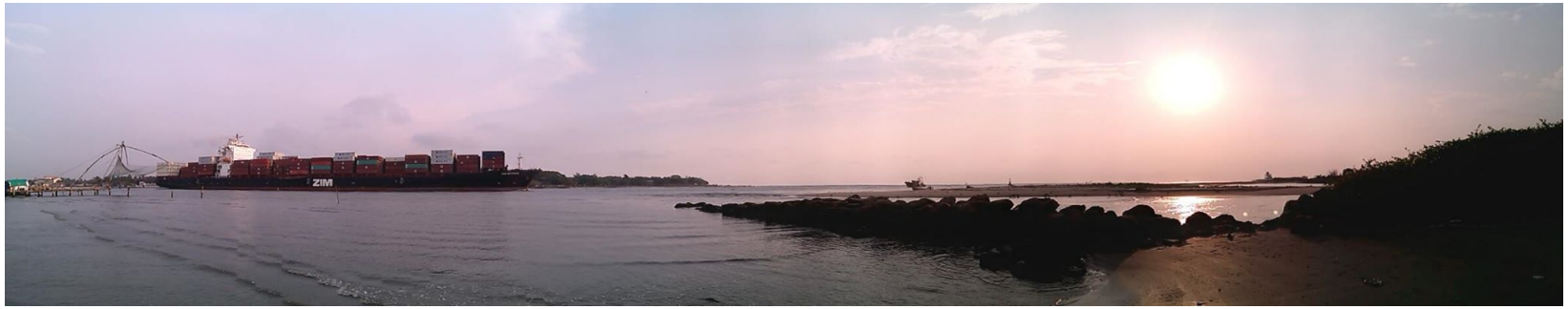
sb = size(b)%get the size of the left image

b= imresize(b,[sa(1) sa(2)]); %now resize 'b' as per the size of 'a' in order to get perfect sized image

c= [a b]; % club the image a & b into another new image after making their size same

%note the semicolon in the above line inside bracket

imshow(c) % show the image


ZIM

Basic Steps:

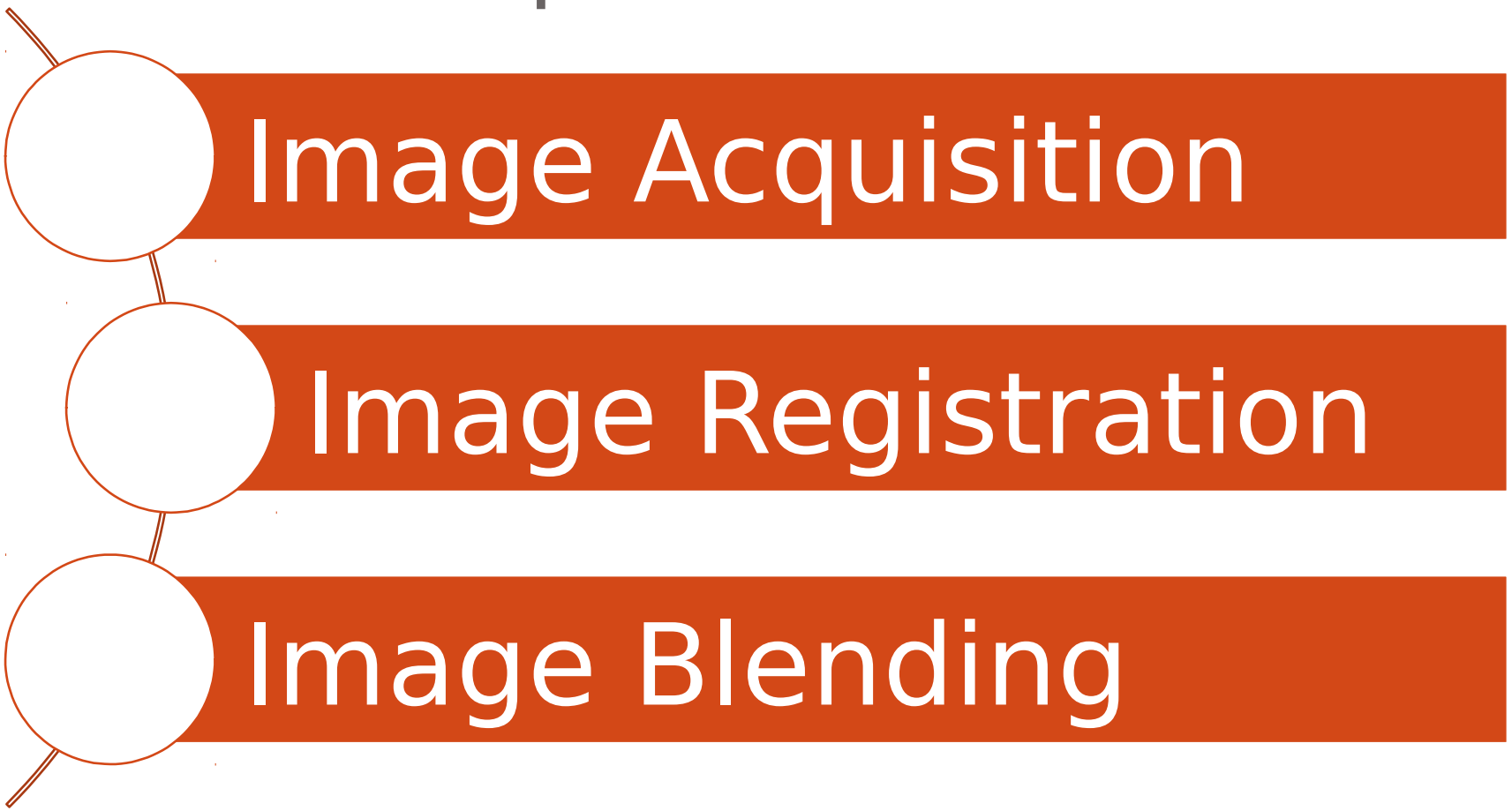


Image Acquisition

- This is the first step
- It is the action of retrieving an image from some sources.
- This can be done using a handheld camera.

Features detection and Matching

- This is considered to be the main step in the image stitching process.
- Features can be defined as the elements in the two or more input images.

There are many algorithms in literature SURF,PCA-SIFT, SIFT, HOG etc. In this we have used SIFT.

SIFT

Scale Invariant Feature Transform

To find the below pictures from this picture:

Let Us Look Into Some Examples

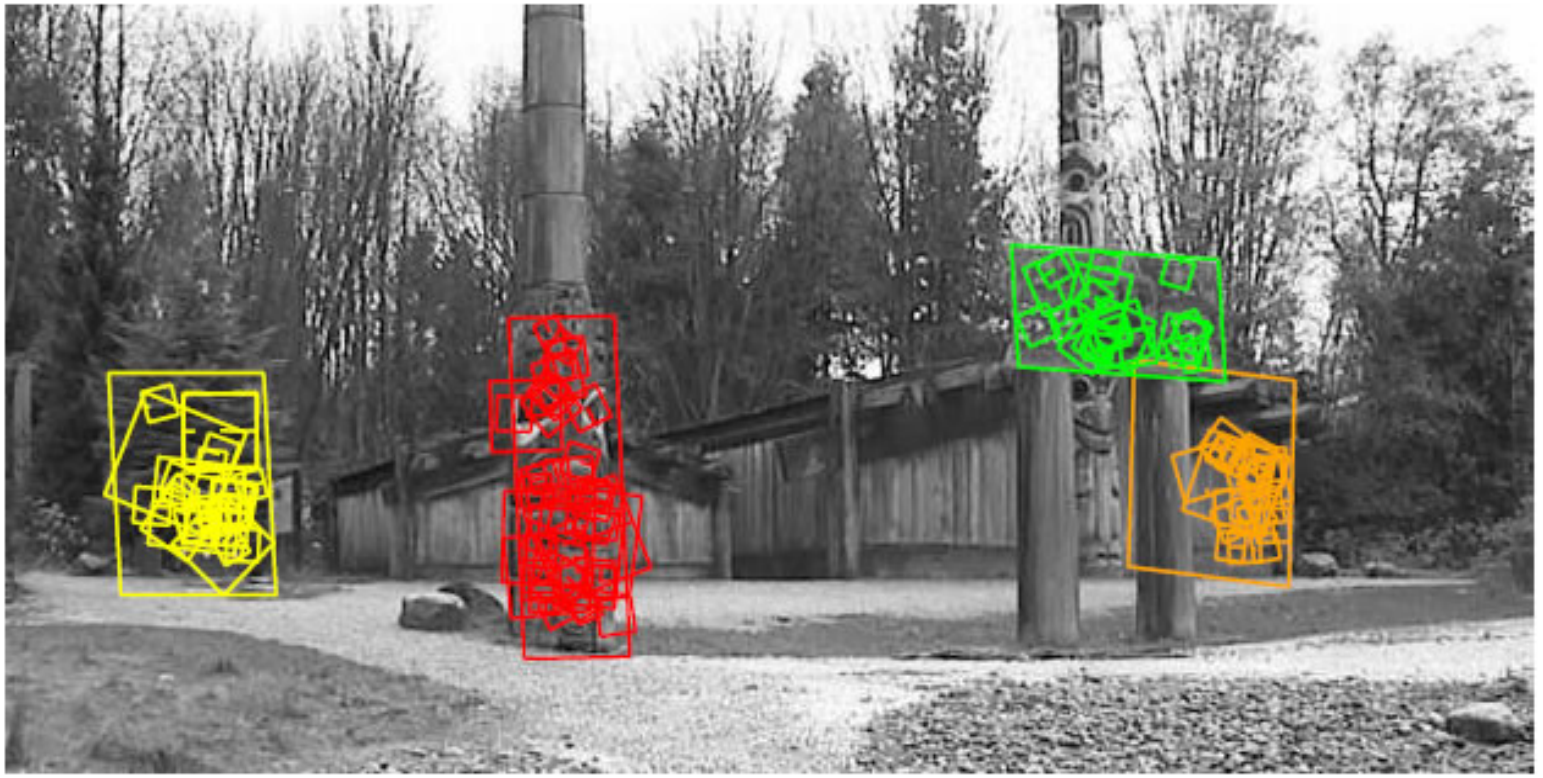
Figure 1

Figure 2

Figure 3

Figure 4

Result!!!!



What the features behave????

- Scale
- Rotation
- Illumination
- Viewpoint

Algorithm...

- **Constructing a scale space**
- **LoG Approximation**
- **Finding keypoints**
- **Get rid of bad key points**
- **Assigning an orientation to the keypoints**
- **Generate SIFT features**

1. Scale Space

Gaussian Blur

- A **Gaussian blur** (also known as Gaussian smoothing) is the result of blurring an image by a Gaussian function.
- It is a widely used effect in graphics software, typically to reduce image noise and reduce detail.
- Gaussian smoothing is also used as a pre-processing stage in computer vision algorithms in order to enhance image structures at different scales

Example

Additional..

- How to make the Gaussian Blur?

Ans : It is same as convolution of the original image with the Gaussian Function.

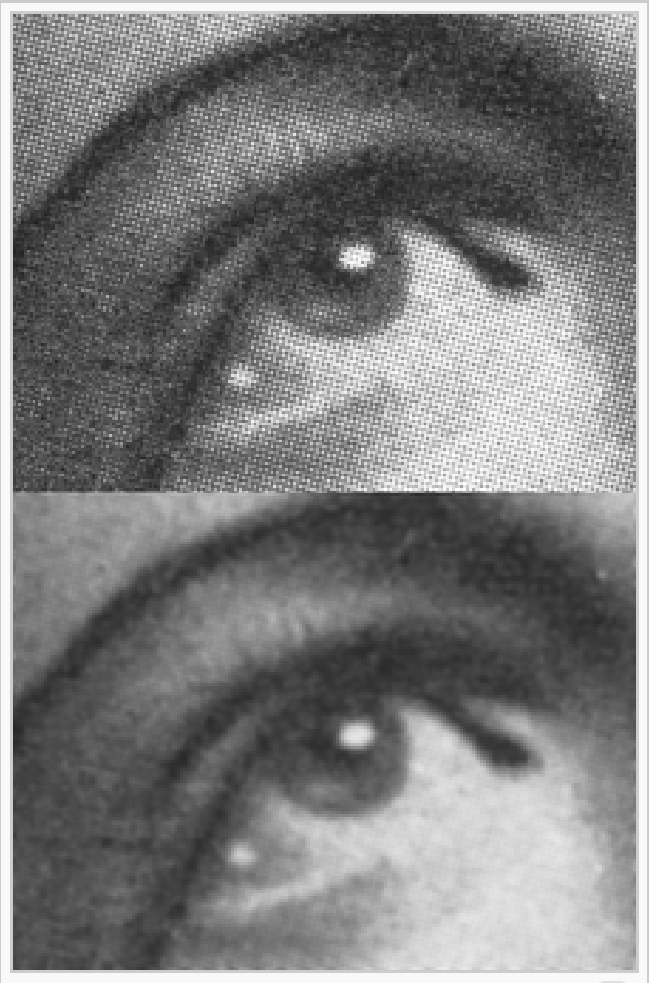
$$G(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}}$$

1 dimension

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

2 dimension

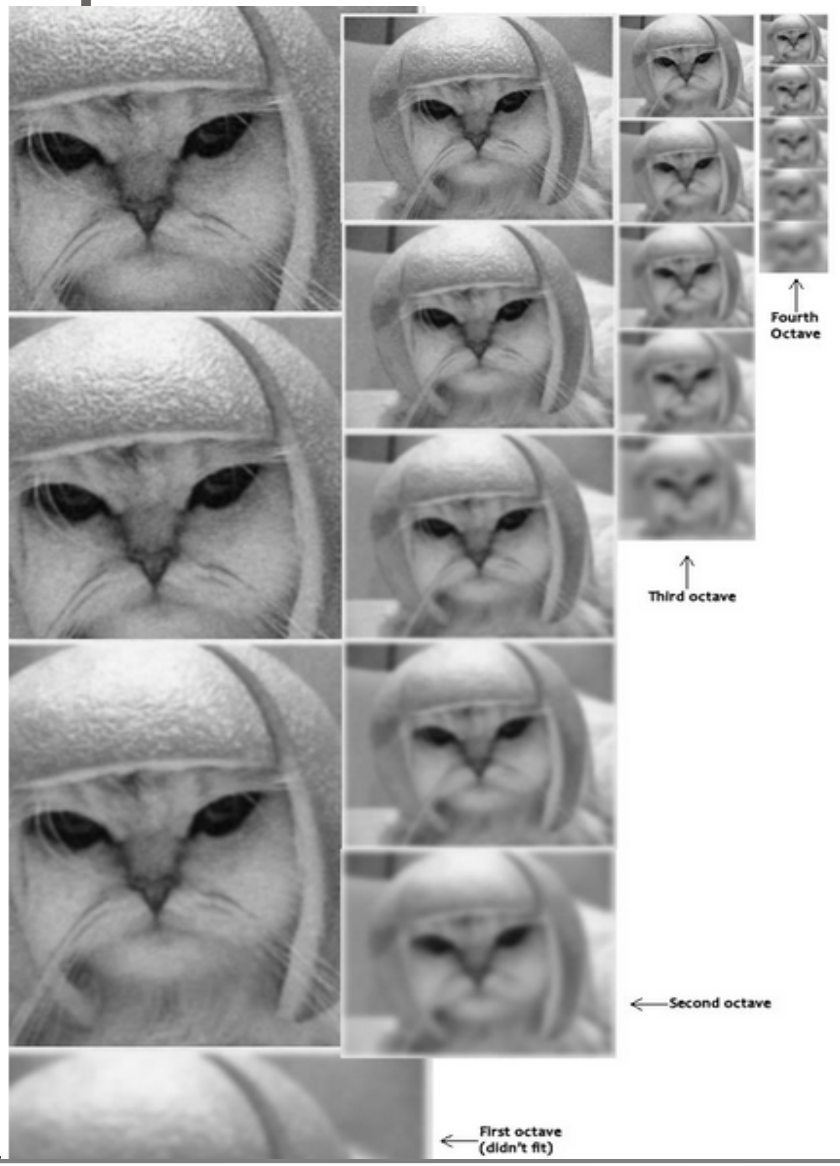
Why Gaussian Blur is used?



Effect of Gaussian Blurring..

- https://upload.wikimedia.org/wikipedia/commons/7/73/Edge_Image.gif

Scale Spaces in SIFT



Octaves and Scales…

- Images seen in the previous slide with the same size(vertical) fall in the same Octave.
- Each octave has different images which is the result of the different standard deviation value or the range of blurring .
- The number of octaves and scale depends on the size of the original image.
- Mostly 4 octaves and 5 blur levels are ideal for the algorithm.

Blurring

- Mathematically, "blurring" is referred to as the convolution of the Gaussian operator and the image.
- Gaussian blur has a particular expression or "operator" that is applied to each pixel. What results is the blurred image.

Cont..

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y)$$

The symbols:

- L is a blurred image
- G is the Gaussian Blur operator
- I is an image
- x,y are the location coordinates
- σ is the "scale" parameter. Think of it as the amount of blur. Greater the value, greater the blur.
- The * is the convolution operation in x and y. It "applies" gaussian blur G onto the image I.

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$$

2. LoG Approximations

- Now we use those blurred images to generate another set of images, the Difference of Gaussians (DoG).
- These DoG images are a great for finding out interesting key points in the image.

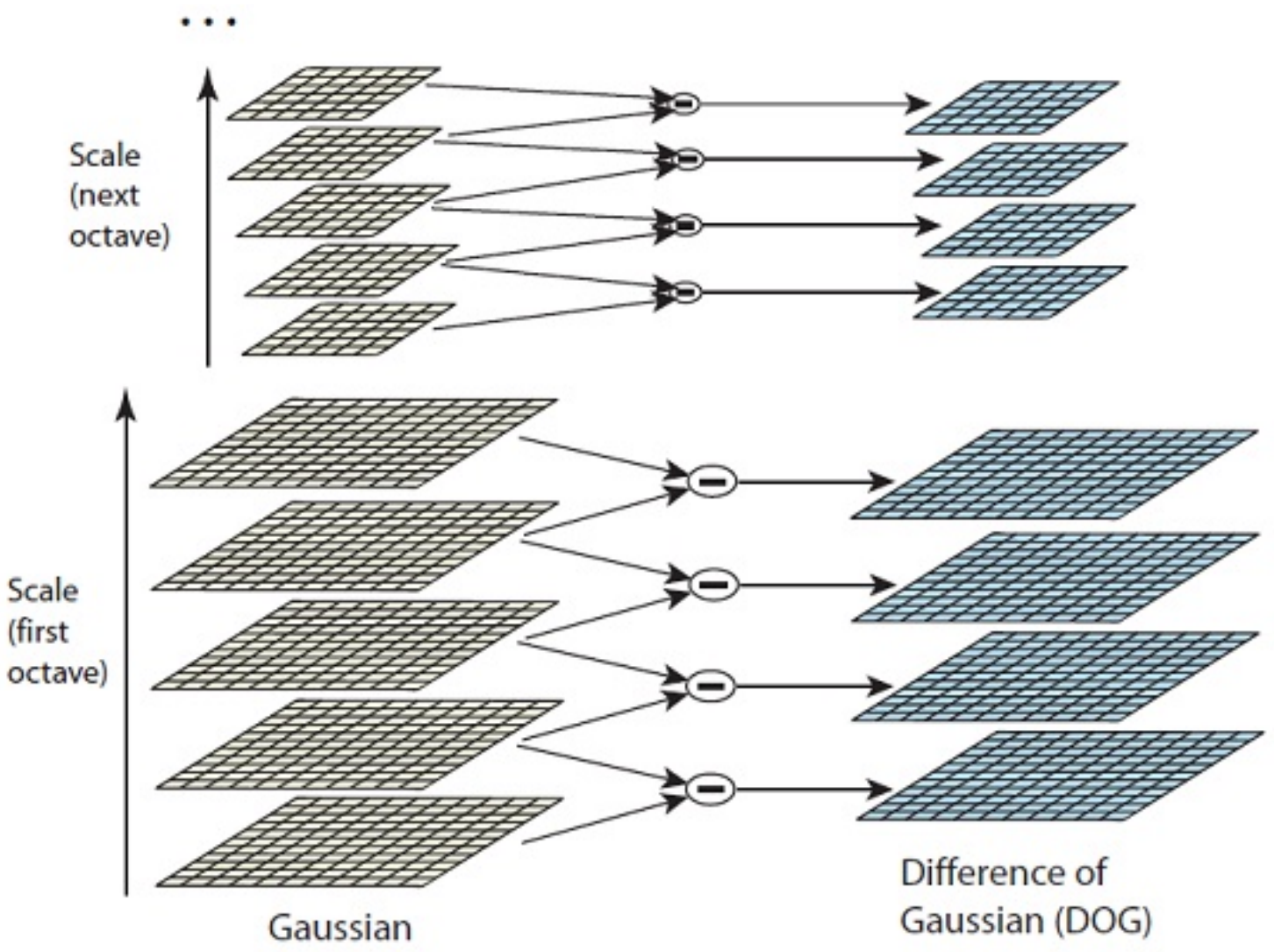
Laplacian of Gaussian (LoG)

Steps:

- You take an image, and blur it a little.
- You calculate second order derivatives on it (or, the "laplacian").
- This locates edges and corners on the image.

Note: These edges and corners are good for finding keypoints.

The Con...


Scale
(next
octave)
Scale
(first
octave)
Gaussian
Difference of
Gaussian (DOG)

Benefits

- Just the Laplacian of Gaussian images are not scale invariant.
- That is, they depend on the amount of blur you d ıe Gaussian expre $G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$

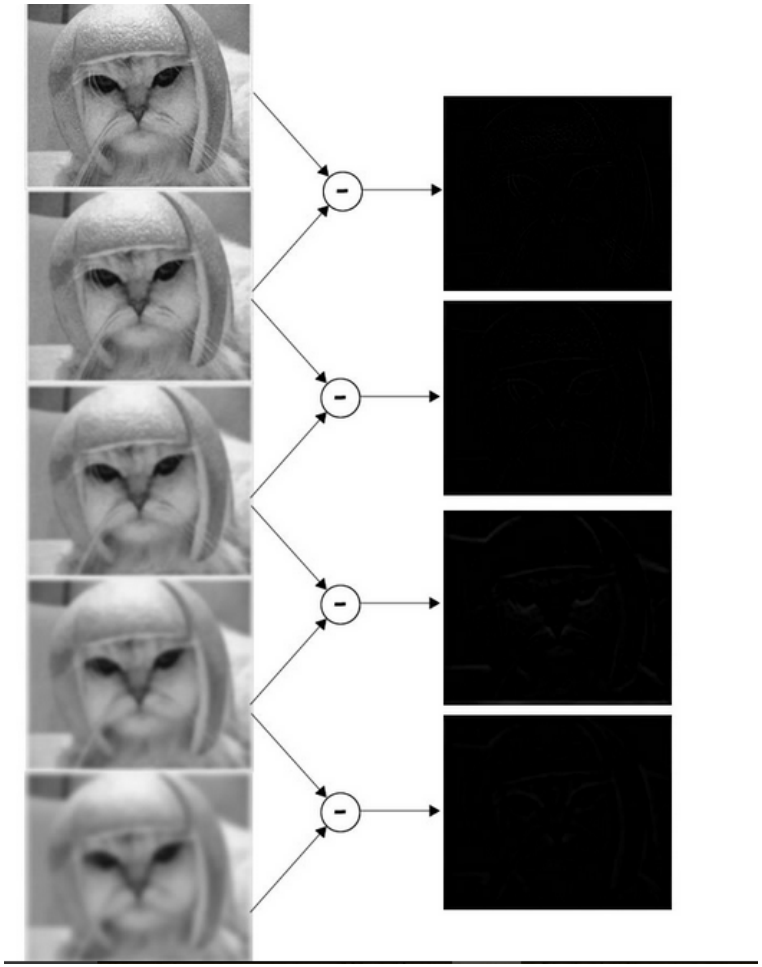
$\nabla^2 G$

$\sigma^2 \nabla^2 G$

- If the LoG is represented by:

Then the scale invariant of LoG :

Example:

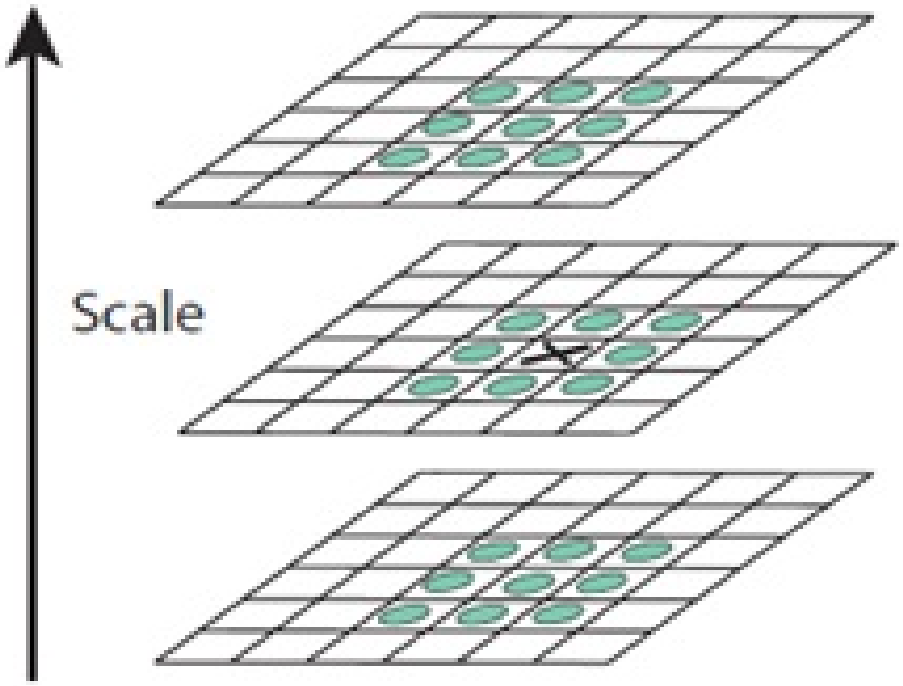


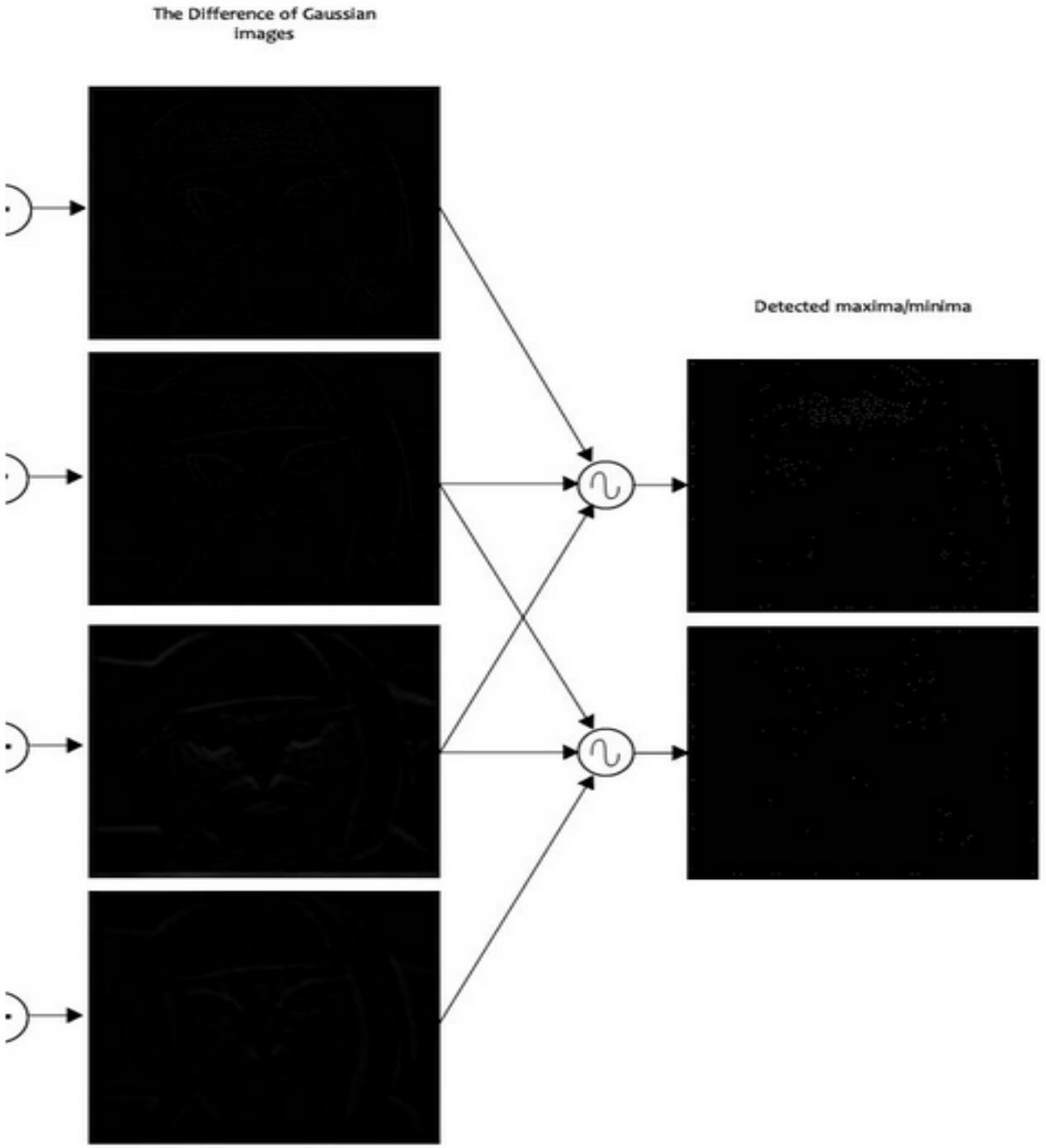
3. Finding Keypoints

Finding the keypoints is a two part process:

- Locate maxima/minima in DoG images
- Find subpixel maxima/minima

Locate maxima/minima in DoG images




The Difference of Gaussian images
Detected maxima/minima

4. Getting rid of low contrast keypoints

- Removing low contrast features:

If the magnitude of the intensity (i.e., without sign) at the current pixel in the DoG image (that is being checked for minima/maxima) is less than a certain value, it is rejected.

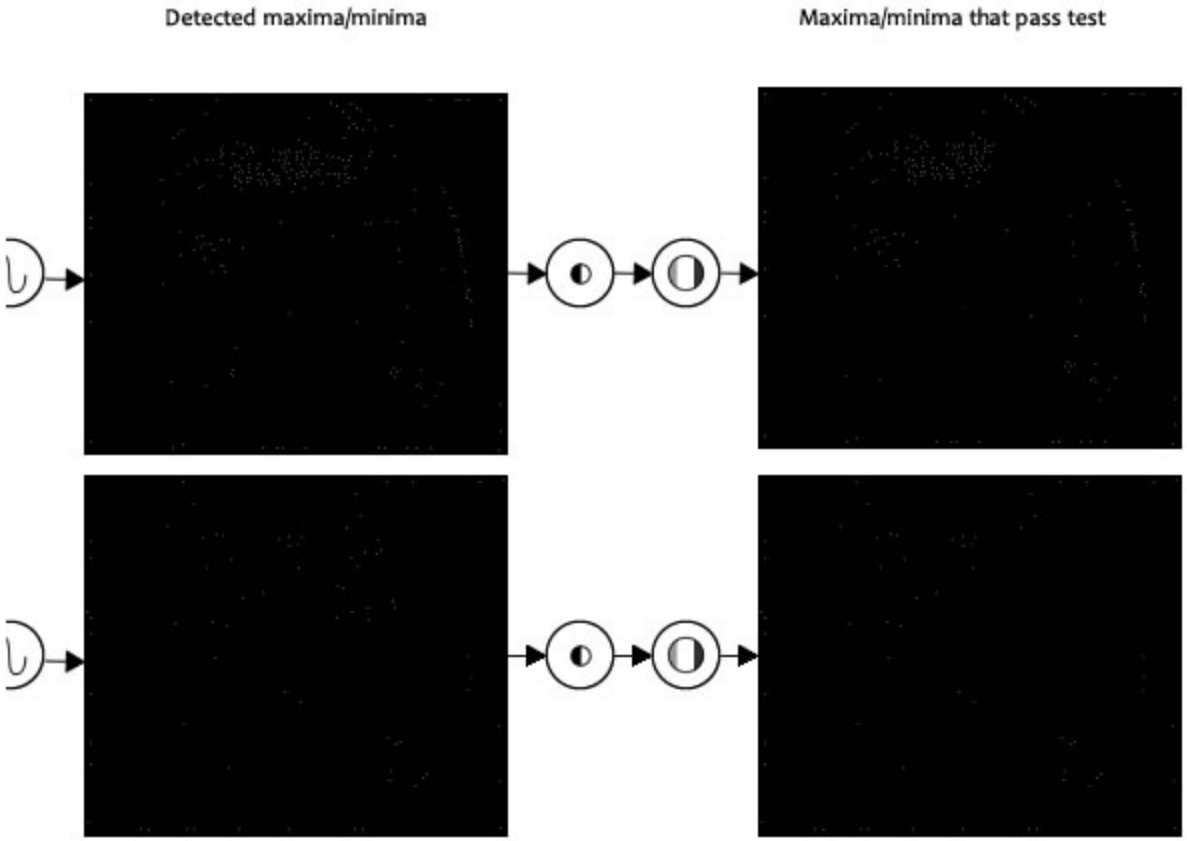
Contd.

- Removing edges:

The idea is to calculate two gradients at the keypoint. Both perpendicular to each other. Based on the image around the keypoint, three possibilities exist. The image around the keypoint can be:

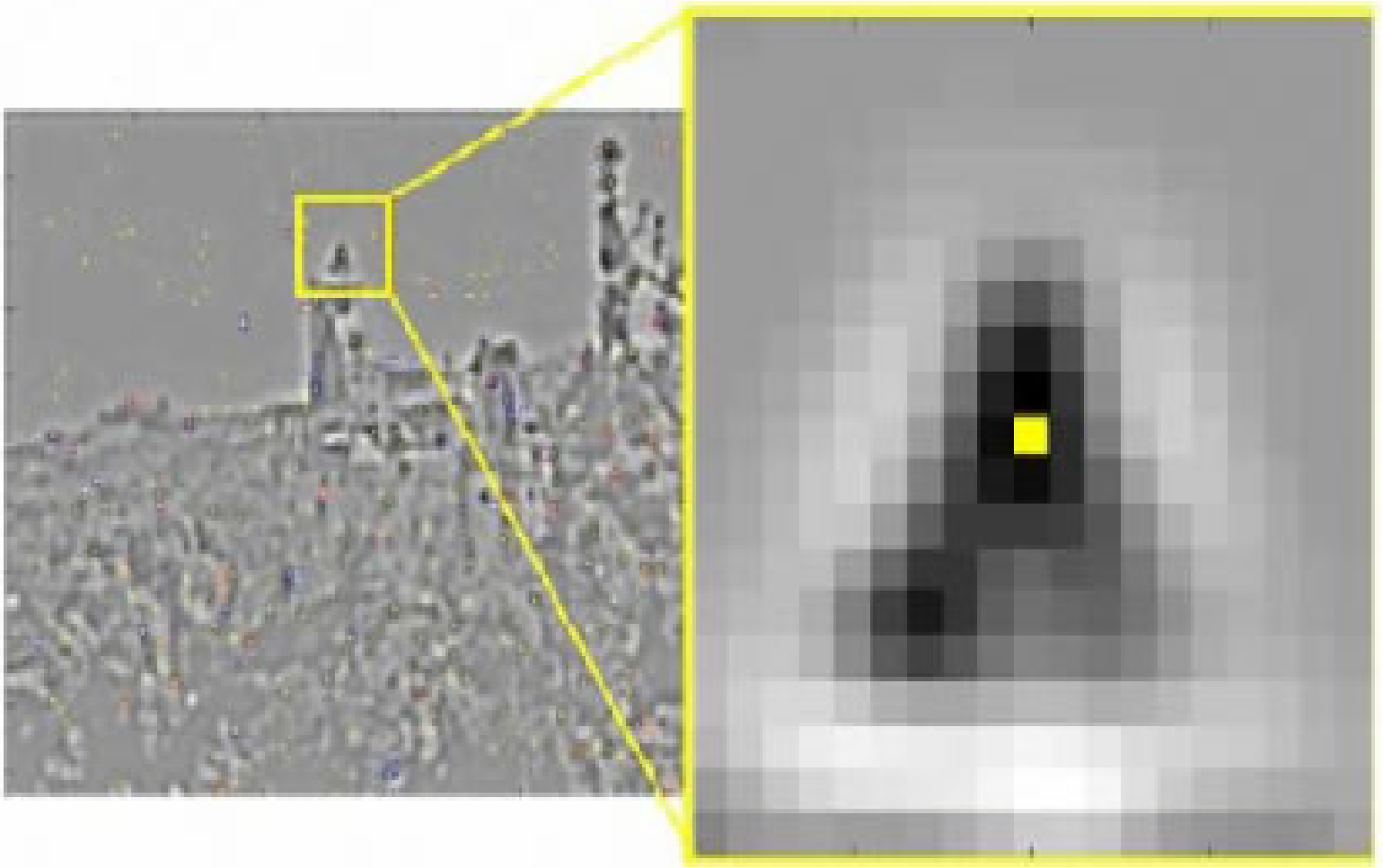
```
* **A flat region:** If this is the case, both gradients will be small.
* **An edge:** Here, one gradient will be big (perpendicular to the edge) and the other will be small (along the edge)
* **A "corner":** Here, both gradients will be big.
```

Visual Explanation:



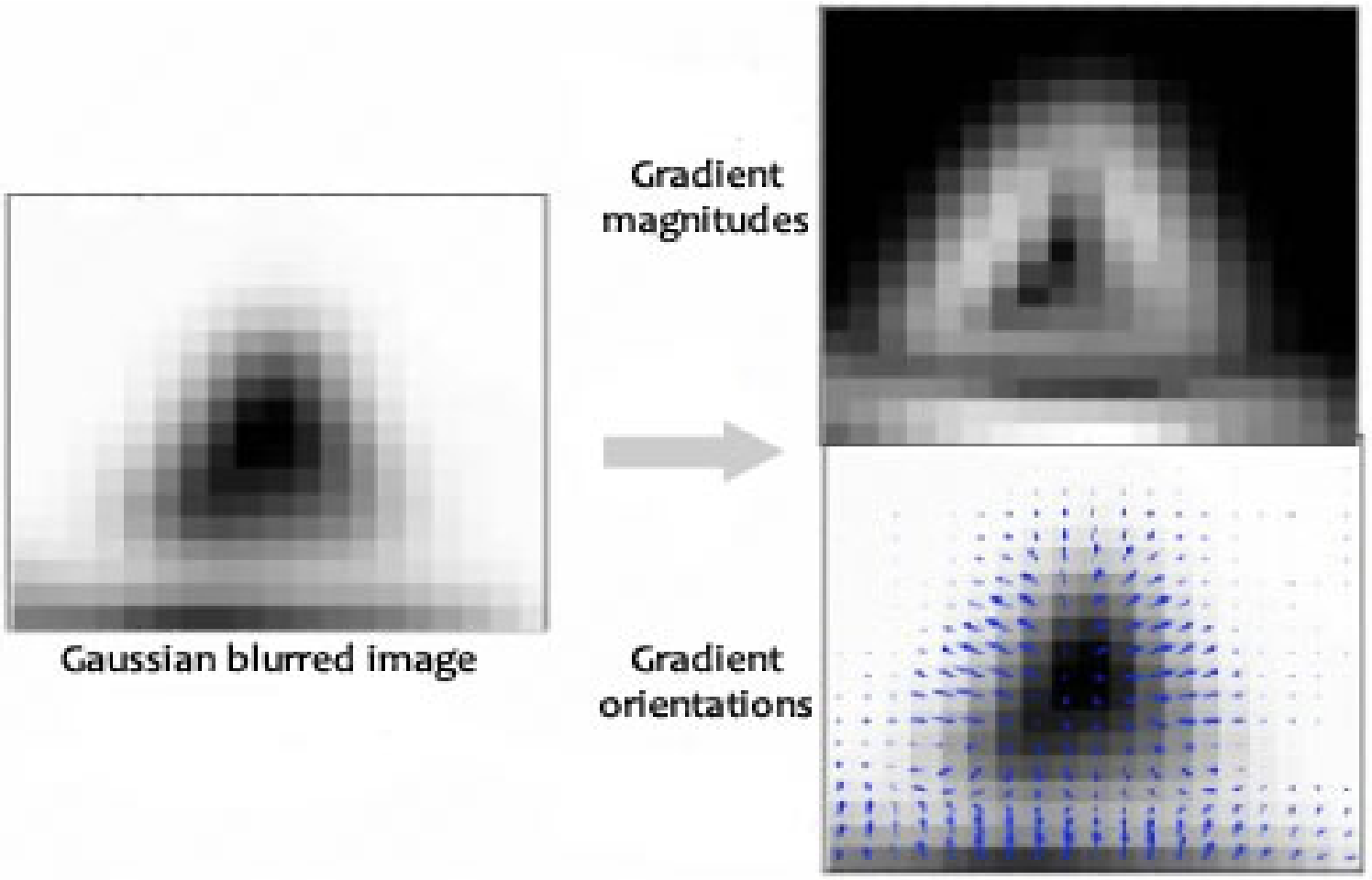
5. Keypoint Orientations

- The idea is to collect gradient directions and magnitudes around each keypoint.
- Then we figure out the most prominent orientation(s) in that region.
- And we assign this orientation(s) to the keypoint.



A keypoint

The Details:

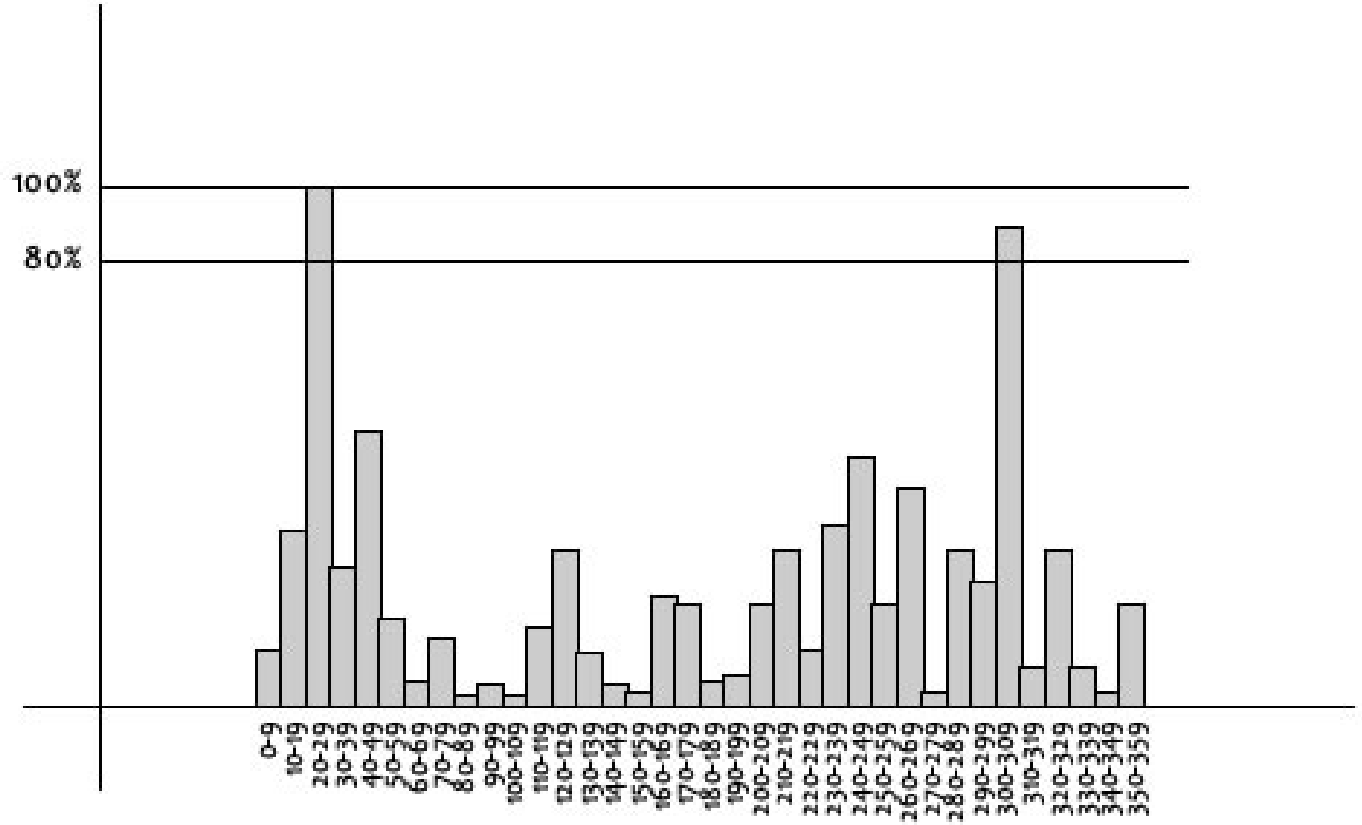

Gradient magnitudes
Gaussian blurred image
Gradient orientations

Gradient magnitudes and orientations formulae:

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2}$$

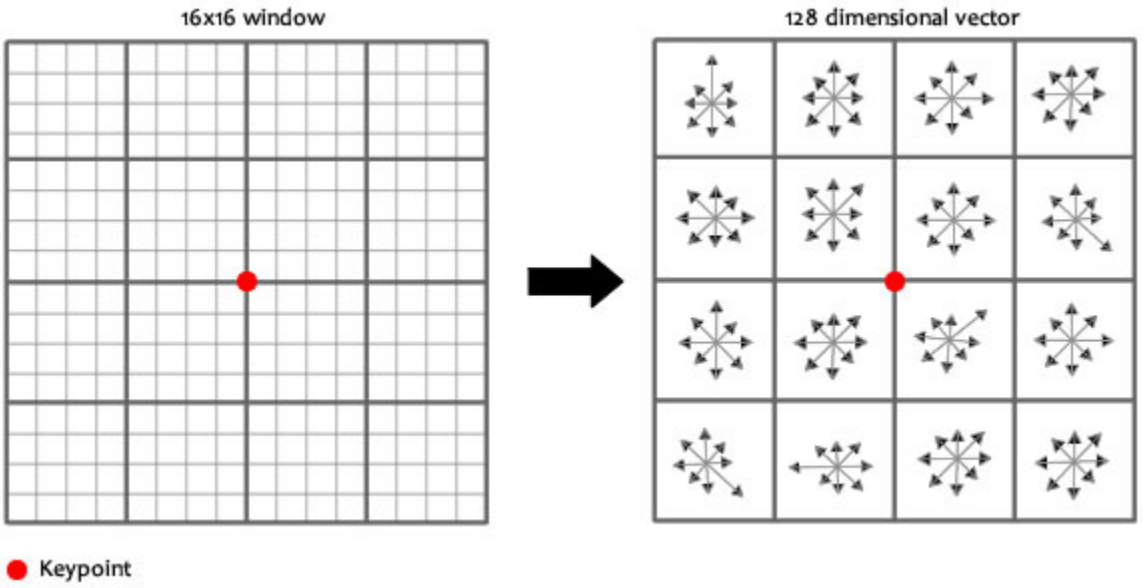
$$\theta(x,y) = \tan^{-1}((L(x,y+1) - L(x,y-1))/(L(x+1,y) - L(x-1,y)))$$

Histogram

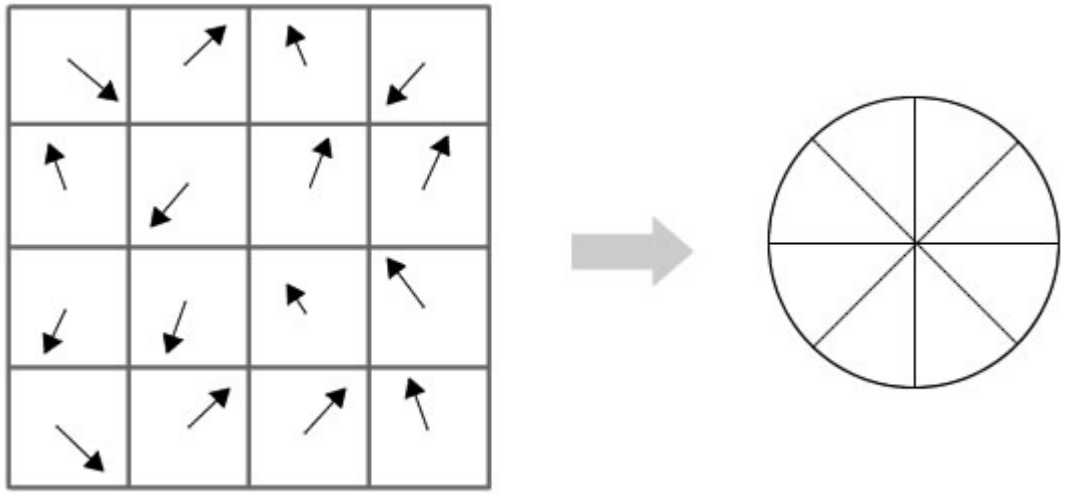

100%
80%
0-9
10-19
20-29
30-39
40-49
50-59
60-69
70-79
80-89
90-99
100-109
110-119
120-129
130-139
140-149
150-159
160-169
170-179
180-189
190-199
200-209
210-219
220-229
230-239
240-249
250-259
260-269
270-279
280-289
290-299
300-309
310-319
320-329
330-339
340-349
350-359

6. Generating a Feature

- This is to make every keypoint a unique.



contd.




Selected Image


Image with key points mapped onto it

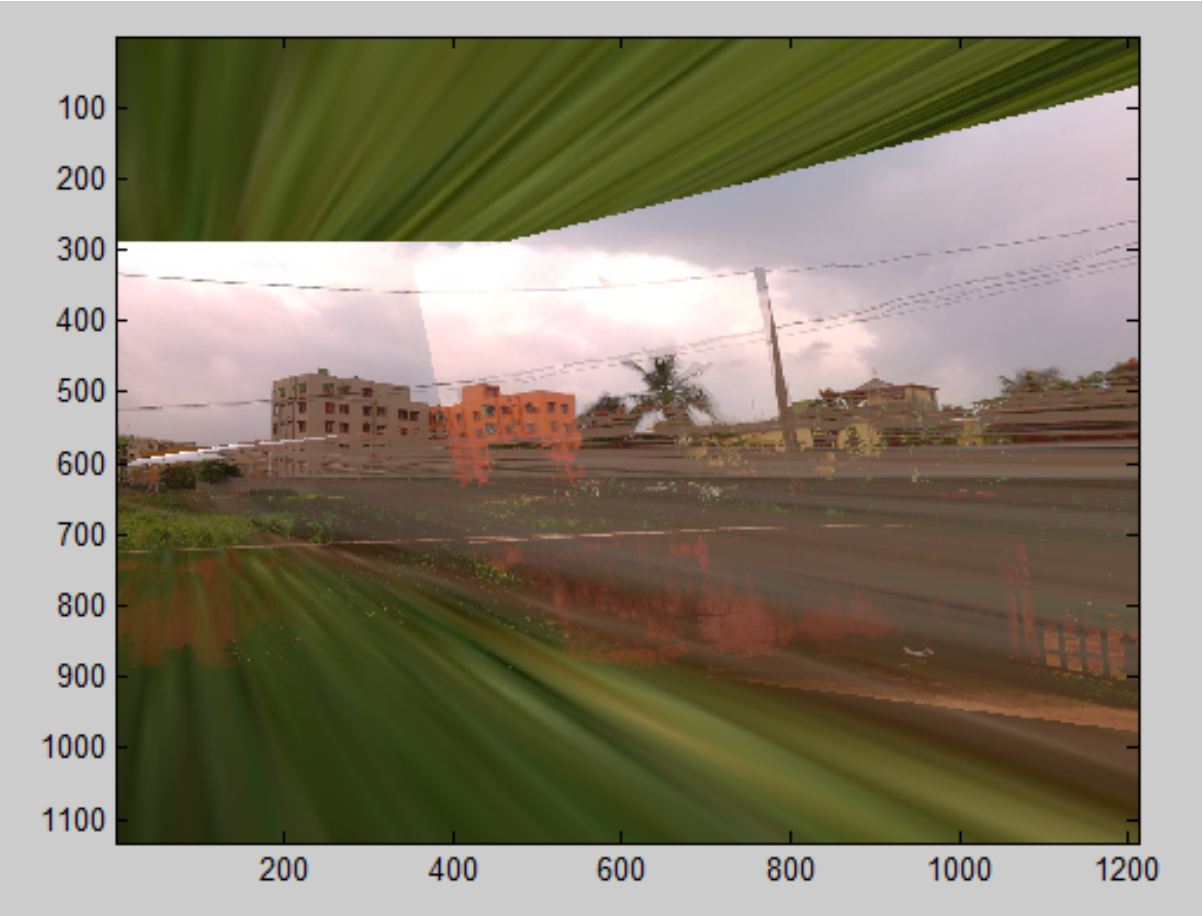

Selected image


Image with key points mapped onto it


Selected image


Image with key points mapped onto it

Final Panorama



References:

- A technical Analysis of Image Stitching Algorithm (Pranoti Kale and KR Singh),IJCSIT,2015,284-288
- Implementation of HDR panorama stitching algorithm (Piotr Ostiak),ICGMS
- http://www.aishack.in/tutorials/sift-scale-invariant-feature-transform-keypoints/

Thank you!!!!