

LEARN OPENCV BY EXAMPLES

OpenCV simplified for beginners by the use of examples. Learn OpenCV with basic implementation of different algorithms.

Home	For Beginners	Table of Contents	Keywords	
----------------------	-------------------------------	-----------------------------------	--------------------------	--

Faster face detection by varying detectMultiScale function parameters

In example 1, a simple face detection code is provided which runs at 1-2 frames per second.

To make it faster, we can modify the parameters to detectMultiScale function to achieve very high frame rate.

Example 2 provides the code for minor optimization of code to make it considerably reducing the size of image for detecting face and remapping the face Region Of Interest (ROI) in the original image.

Here the last two parameters - *Size minSize=Size()* and *Size maxSize=Size()* - are adaptively changed to reduce computational complexity. These parameters are responsible for **restricting the search window of the function to vary from minimum size(minSize) to maximum size(maxSize)** for face detection.

Note : You need a camera connected, otherwise it will throw an error.

Example 1: (Face detection from Video captured from camera)

```
1 | #include "opencv2/objdetect/objdetect.hpp"
```

SEARCH CONTENTS OF THIS BLOG

POPULAR POSTS

- 1 Find Contour
- 2 Basic drawing examples
- 3 Line Detection by Hough Line Transform
- 4 Face Detection using Haar-Cascade Classifier
- 5 Perspective Transform
- Sobel Edge Detection

```

2  #include "opencv2/highgui/highgui.hpp"
3  #include "opencv2/imgproc/imgproc.hpp"
4
5  #include <iostream>
6  #include <stdio.h>
7
8  using namespace std;
9  using namespace cv;
10
11 Mat detectFace(Mat src);
12
13 int main( )
14 {
15     VideoCapture cap(0);
16     namedWindow( "window1", 1 );
17
18     while(1)
19     {
20         Mat frame;
21         cap >> frame;
22         frame=detectFace(frame);
23
24         imshow( "window1", frame );
25         // Press 'c' to escape
26         if(waitKey(1) == 'c') break;
27     }
28
29     waitKey(0);
30     return 0;
31 }
32
33 Mat detectFace(Mat image)
34 {
35     // Load Face cascade (.xml file)
36     CascadeClassifier face_cascade( "C:/OpenCV243/data/Haarcascades/ha
37
38     // Detect faces
39     std::vector<Rect> faces;
40     face_cascade.detectMultiScale( image, faces, 1.1, 2, 0|CV_HAAR_SCA
41
42     // Draw circles on the detected faces
43     for( int i = 0; i < faces.size(); i++ )
44     {
45         Point center( faces[i].x + faces[i].width*0.5, faces[i].y + faces
46         ellipse( image, center, Size( faces[i].width*0.5, faces[i].height
47     }
48     return image;
49 }

```

Kalman Filter Implementation (Tracking mouse position)

Histogram Calculation

OpenCV example to convert RGB to gray / other color spaces

Hough Circle Detection


CATEGORIES

- [Accessory](#)
- [Applications](#)
- [Basics](#)
- [Edge Detection](#)
- [Feature Extraction](#)
- [Filter](#)
- [Miscellaneous](#)
- [Morphological Operation](#)

Example 2: (Optimizing the parameters of detectMultiScale for faster operation)

```
1  #include "opencv2/objdetect/objdetect.hpp"
2  #include "opencv2/highgui/highgui.hpp"
3  #include "opencv2/imgproc/imgproc.hpp"
4
5  #include <iostream>
6  #include <stdio.h>
7
8  using namespace std;
9  using namespace cv;
10
11  double min_face_size=20;
12  double max_face_size=200;
13
14  Mat detectFace(Mat src);
15
16  int main( )
17  {
18      VideoCapture cap(0);
19      namedWindow( "window1", 1 );
20
21      while(1)
22      {
23          Mat frame;
24          cap >> frame;
25          frame=detectFace(frame);
26
27          imshow( "window1", frame );
28          // Press 'c' to escape
29          if(waitKey(1) == 'c') break;
30      }
31
32      waitKey(0);
33      return 0;
34  }
35
36  Mat detectFace(Mat image)
37  {
38      // Load Face cascade (.xml file)
39      CascadeClassifier face_cascade( "C:/OpenCV243/data/Haarcascades/"
40
41      // Detect faces
42      std::vector<Rect> faces;
43
44      face_cascade.detectMultiScale( image, faces, 1.2, 2, 0|CV_HAAR_S
45
46      // Draw circles on the detected faces
47      for( int i = 0; i < faces.size(); i++ )
48      {
49          min_face_size = faces[0].width*0.8;
50          max_face_size = faces[0].width*1.2;
51          Point center( faces[i].x + faces[i].width*0.5, faces[i].y + fac
```

```
52 | ellipse( image, center, Size( faces[i].width*0.5, faces[i].height  
53 | }  
54 | return image;  
55 | }
```

 Recommend this on Google

Labels: [Applications](#)

2 comments:



Takeyasu Nakazato February 4, 2014 at 6:43 PM

Thank you so much. thank you. But i have one question: How to crop the ellipse area?

[Reply](#)

[Replies](#)



Kanha February 7, 2014 at 2:39 PM

please visit this page for your answer...

<http://opencvexamples.blogspot.com/2014/02/crop-elliptical-region-from-image.html>

[Reply](#)