



Tracking colored objects in OpenCV

By [Utkarsh](#) | Published: July 19, 2010

Suka 2.575 orang menyukai ini.

If you're new to image processing, you'll enjoy this project. What we'll attempt to achieve in this tutorial is tracking the location of a coloured object in an image. In our case, it'll be a yellow colored ball. Once we're finished, we'll have something like this:

Looks great right? So lets dive right in!

Creating the project

First, create a Win32 console application. Choose any name you like, and accept the default wizard options. You'll get an empty project with a main function. First, add these header files to the code:

```
#include "cv.h"
#include "highgui.h"
```

Next, add the library files to the project. Go to Project > Properties > Configuration > Linker > Input and write cv.lib cxcore.lib cvaux.lib highgui.lib in Additional Dependencies.

If you have any problems with setting this up, I suggest you go through [Using OpenCV with Window](#).

The plan of action

Before diving right into the code, its always a good idea to put a little insight into what we're doing. Our program flow should go something like this:

- Get an image from the camera
- Figure out where the yellow ball is
- Add the current position to an array of some sort

To get an image from the camera, we'll use code from [Capturing Images](#), that is, we'll use inbuilt OpenCV functions that let you access camera.

For figuring out where the ball is, we'll first threshold the image and use zero order and first order moments.

To keep a track of where the ball has been, we'll use another image. We'll keep drawing wherever the ball goes, and combine this

image with the original frame. That way, we'll get a "scribble" like effect. You'll see what I mean when we implement it in code.

The dive into code

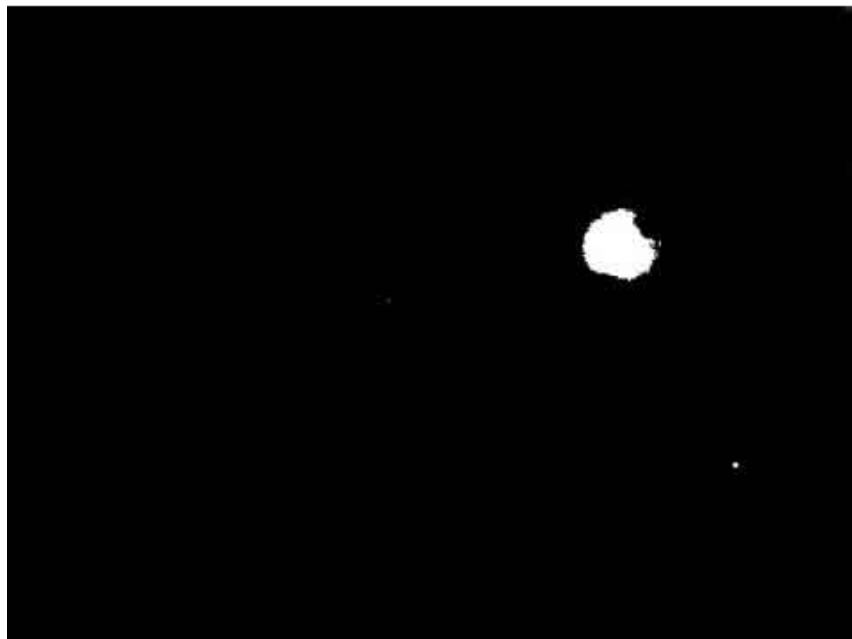
We'll start off by writing the thresholding function:

```
IplImage* GetThresholdedImage(IplImage* img)
{
```

This function will take an image, and return a binary image (where yellow will be white and the rest will be black). Here's a sample of what might be a scenario:



Thresholding



To achieve this thresholding, we'll be using the HSV colour space, instead of the more common RGB colour space. In HSV, each "tint" of colour is assigned a particular number (the Hue). The "amount" of colour is assigned another number (the Saturation) and the brightness of the colour is assigned another number (the Intensity or Value).

This gives us the advantage of having a single number (hue) for the yellow ball despite multiple shades of yellow (all the way from

dark yellow to a bright yellow). For more information you might want to read up [Colour spaces – Grayscale, RGB, HSV and Y'CrCb](#).

Back to the code now. Firstly, we convert the image into an HSV image:

```
// Convert the image into an HSV image
IplImage* imgHSV = cvCreateImage(cvGetSize(img), 8, 3);
cvCvtColor(img, imgHSV, CV_BGR2HSV);
```

We keep the original image (img) intact, for future uses. The image is originally stored in the BGR format, so we convert BGR into HSV.

Now, create a new image that will hold the thresholded image (which will be returned).

```
IplImage* imgThreshed = cvCreateImage(cvGetSize(img), 8, 1);
```

Now we do the actual thresholding:

```
cvInRangeS(imgHSV, cvScalar(20, 100, 100), cvScalar(30, 255, 255), imgThreshed);
```

Here, imgHSV is the reference image. And the two cvScalars represent the lower and upper bound of values that are yellowish in colour. (These bounds should work in almost all conditions. If they don't, try experimenting with the last two values).

Consider any pixel. If all three values of that pixel (H, S and V, in that order) lie within the stated ranges, imgThreshed gets a value of 255 at that corresponding pixel. This is repeated for all pixels. So what you finally get is a thresholded image.

And finally, release the temporary HSV image and return this thresholded image:

```
cvReleaseImage(&imgHSV);
return imgThreshed;
}
```

That finishes up our thresholding function.

Next we'll get to the main function:

```
int main()
{
```

First, we initialize the capturing device. If we don't get a device, we simply exit... no questions asked.

```
// Initialize capturing live feed from the camera
CvCapture* capture = 0;
capture = cvCaptureFromCAM(0);

// Couldn't get a device? Throw an error and quit
if(!capture)
{
    printf("Could not initialize capturing...\n");
    return -1;
}
```

And then we setup windows that will display the live images:

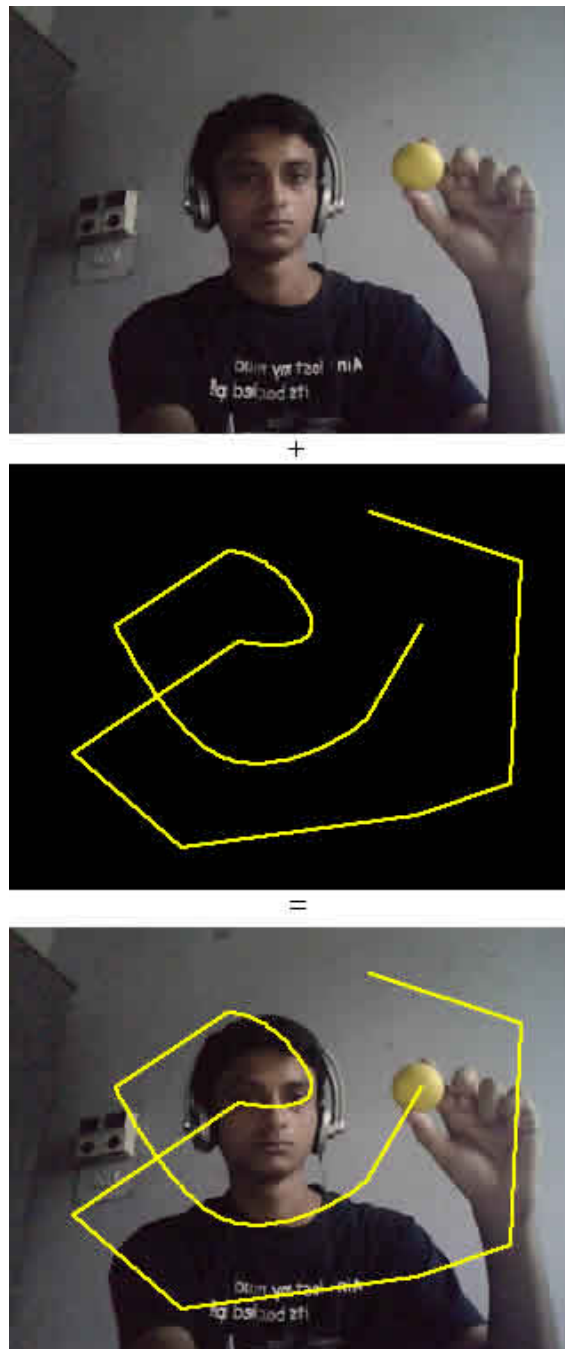
```
// The two windows we'll be using
cvNamedWindow("video");
cvNamedWindow("thresh");
```

video will display the actual output of the program (like the one you saw in the video at the top of this image). thresh will display the thresholded image, just to aid debugging if its needed.

Now we initialize the image that will hold the "scribble" data.

```
// This image holds the "scribble" data...
// the tracked positions of the ball
IplImage* imgScribble = NULL;
```

We'll keep updating imgScribble with appropriate lines. And we'll add this image to the current frame.. and we'll get the final output.
Here's a possible situation:



I hope it makes sense.

Moving on, we create an infinite loop (we're working on a realtime project here):

```
// An infinite loop
while(true)
{
    // Will hold a frame captured from the camera
    IplImage* frame = 0;
    frame = cvQueryFrame(capture);
```

We capture a frame from the camera, and store it in frame.

Just in case, if we won't get a frame, we simply quit.

```
// If we couldn't grab a frame... quit
if(!frame)
    break;
```

If you noticed, we just created `imgScribble`. We didn't allocate any memory to it. The first frame would be a good place to do so. And to determine if its the first frame, we can check if `imgScribble` is currently `NULL` or not:

```
// If this is the first frame, we need to initialize it
if(imgScribble == NULL)
{
    imgScribble = cvCreateImage(cvGetSize(frame), 8, 3);
}
```

If the code reaches this far, we're sure that a frame was captured, and the `imgScribble` is a valid image. So we get down to business, and generate the thresholded image using the function we wrote above:

```
// Holds the yellow thresholded image (yellow = white, rest = black)
IplImage* imgYellowThresh = GetThresholdedImage(frame);
```

Now `imgYellowThresh` holds a binary image similar to the ones shown above. Now we use mathematics based calculations to figure out the position of the yellow ball.

NOTE: I'm assuming that there will be only one yellow object on screen. If you have multiple objects, this code won't work.

```
// Calculate the moments to estimate the position of the ball
CvMoments *moments = (CvMoments*)malloc(sizeof(CvMoments));
cvMoments(imgYellowThresh, moments, 1);

// The actual moment values
double moment10 = cvGetSpatialMoment(moments, 1, 0);
double moment01 = cvGetSpatialMoment(moments, 0, 1);
double area = cvGetCentralMoment(moments, 0, 0);
```

You first allocate memory to the moments structure, and then you calculate the various moments. And then using the moments structure, you calculate the two first order moments (`moment10` and `moment01`) and the zeroth order moment (`area`).

Dividing `moment10` by `area` gives the X coordinate of the yellow ball, and similarly, dividing `moment01` by `area` gives the Y coordinate.

Now, we need some mechanism to be able to store the previous position. We do that using static variables:

```
// Holding the last and current ball positions
static int posX = 0;
static int posY = 0;

int lastX = posX;
int lastY = posY;

posX = moment10/area;
posY = moment01/area;
```

The current position of the ball is stored in `posX` and `posY`, and the previous location is stored in `lastX` and `lastY`.

We'll just print out the current position for debugging purposes:

```
// Print it out for debugging purposes
printf("position (%d,%d)\n", posX, posY);
```

Now, we do some scribbling:

```
// We want to draw a line only if its a valid position
if(lastX>0 && lastY>0 && posX>0 && posY>0)
{
    // Draw a yellow line from the previous point to the current point
    cvLine(imgScribble, cvPoint(posX, posY), cvPoint(lastX, lastY), cvScalar(0,255,255), 5);
}
```

We simply create a line from the previous point to the current point, of yellow colour and a width of 5 pixels.

The if condition prevents any invalid points from being drawn on the screen. (Just try taking the yellow object out of the screen once the program is done... you'll see what I mean).

Once all of this processing is over, we combine the scribble and the captured frame:

```
// Add the scribbling image and the frame...
cvAdd(frame, imgScribble, frame);
cvShowImage("thresh", imgYellowThresh);
cvShowImage("video", frame);
```

After displaying the images, we check if a key was pressed:

```
// Wait for a keypress
int c = cvWaitKey(10);
if(c!=-1)
{
    // If pressed, break out of the loop
    break;
}
```

If a key was pressed, break out of the loop.

And finally, we release the thresholded image. We don't want to accumulate multiple thresholded images...

```
// Release the thresholded image+moments... we need no memory leaks.. please
cvReleaseImage(&imgYellowThresh);
delete moments;
}
```

And finally, once the loop gets over, we release the camera so that other program can use it:

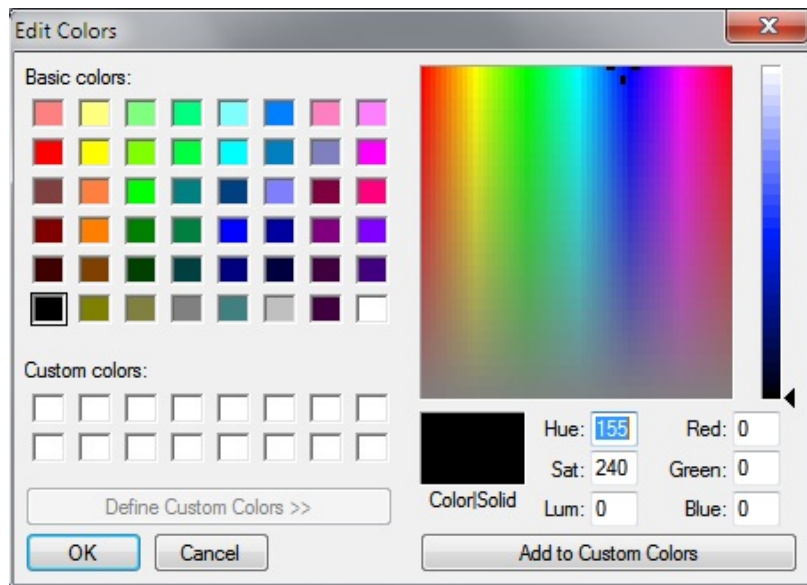
```
// We're done using the camera. Other applications can now use it
cvReleaseCapture(&capture);
return 0;
}
```

Thats it! Try running the program now, it should work, just like in the video!

Tracking different colors

If you want to try some different color, you'll have to figure out it's hue. There are two ways to do that. First – hit and trial. Go through all possible values and you'll hopefully end up getting a good value.

The other method requires using some photo manipulation software (MS Paint will do). Open the color selection palette. Go through the colors and you should see a text box labeled Hue.



Go through all possible Hues to find the range of values. For example, in MS Paint, it is 0-239. But OpenCV's hue values range from 0-179. So you need to scale any hue value you take from MS Paint (multiple the hue from MS Paint by 180/240).

Wrap up

Hope you learned something from this little project! Got any questions? Criticism? Suggestions? **Leave a comment**, or [contact me](#).

Issues? Suggestions? Visit the [Github issue tracker for AI Shack](#)

Back to top

Suka 2.575 orang menyukai ini.

Downloads

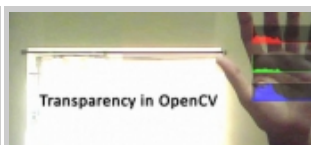
git [Get the code at GitHub](#)

Summary

- 💡 To track a single blob, you can use moments.
- 💡 The tracking can be shaky, depending on the camera quality
- 💡 The HSV colour space can be helpful when segmenting based on colour.



[Capturing images](#)



[Transparent image overlays in OpenCV](#)



[HighGUI: Creating Interfaces](#)

174 Comments



shashi kiran

Posted April 22, 2011 at 10:13 pm | [Permalink](#)

dude plzzzz help me to know the pixel position of that yellow threshed image,where that yellow ball becomes white after applying thresholded image function.plzzzz give me the code for that pixel position of that white space

[Reply](#)



Utkarsh

Posted April 29, 2011 at 9:13 pm | [Permalink](#)

Sorry, didn't get you.

[Reply](#)



shashikiran

Posted May 16, 2011 at 9:08 pm | [Permalink](#)

dude plz help me in getting pixel position(x,y) of that yellow ball in ur thresholded image.plz upload me the code for the above problem.

[Reply](#)



Donny

Posted April 27, 2011 at 7:22 am | [Permalink](#)

What does it mean by lower bound and upper bound? Lower bound for??upper bound for??
Thank you.

[Reply](#)



Utkarsh

Posted April 29, 2011 at 9:05 pm | [Permalink](#)

There's an upper bound and a lower bound – the upper limit and the lower limit. Anything in between in okay.

[Reply](#)



KaiL

Posted April 27, 2011 at 7:59 am | [Permalink](#)

How am i going to do so ?? Would you mind posting the source code . Thank you .

[Reply](#)



Buddy

Posted May 9, 2011 at 12:18 pm | [Permalink](#)

I've tried it and runs successfully. 😊
What should I do if I want to detect two objects (yellow and green color) ???

I wanna change the cvLine drawing with cvCircle without holding the last drawing position.
Could you tell me how?

Thx...

[Reply](#)



Kitty

Posted May 11, 2011 at 1:58 pm | [Permalink](#)

Your program basic on Visual c++ 2008? OR 2005? OR.....?

I couldn't use it and couldn't read stdafx.h, (*error, no such file direction). Do you know what happened? I using 2008 VC++

[Reply](#)



Utkarsh

Posted June 17, 2011 at 4:09 pm | [Permalink](#)

Just remove that line. Or create a blank file with that name. Everything should work then.

[Reply](#)



Pondit Bhaia

Posted May 21, 2011 at 4:21 am | [Permalink](#)

Hello AiShack,

Nice explained. But the problem is you explained it too much. And at the end this episode example source code is not written, so that we can copy that and make more updates on your lecture.

Would it be possible to file your source code at the end of your lecture, as a reader we can do more nicely follow up, and provide more ideas and feedbacks?

Regards
Pondit Bhaia

[Reply](#)



bryan

Posted May 23, 2011 at 11:20 am | [Permalink](#)

error C2601: 'GetThresholdedImage' : local function definitions are illegal
how do i go about solving this?

[Reply](#)



Utkarsh

Posted June 17, 2011 at 4:22 pm | [Permalink](#)

That sounds like a C/C++ problem. Figure it out yourself!

[Reply](#)



Manindra

Posted May 29, 2011 at 12:59 am | [Permalink](#)

Hey guys, for detecting any particular colour, as Utkarsh has said, you need its particular HSV value range. It's difficult doing it in paint. ColorPic is a nifty free tool that can help you determine HSV values, Link: <http://www.iconico.com/download.aspx?type=free> ColorPic has hue values in the range of 0-360. So you'll need to divide it by 2 before using in openCV.

P.S: This one's a superb tutorial! Loved it. Learning from it, I was able to implement colour object tracking in openFrameworks. oF makes things simpler. The complexity of malloc function can be avoided by using the contourFinder class. Comment and let me know if you me to share the code. 😊

[Reply](#)



Utkarsh

Posted June 17, 2011 at 4:21 pm | [Permalink](#)

Thanks for sharing!

Yes, oF does make things a lot simpler. But if you're into hardcore computer vision, you'll want to with directly with OpenCV!

[Reply](#)



Manindra

Posted June 19, 2011 at 12:20 pm | [Permalink](#)

I made a simple Air Hockey game as proof of concept. It uses a thresholding technique similar to yours. Check out the video link at: <http://www.youtube.com/watch?v=vKyWIMFRLKY> 😊

[Reply](#)



Utkarsh

Posted June 21, 2011 at 12:14 pm | [Permalink](#)

Looks great! Just tweeted about it!

[Reply](#)



varan1

Posted June 4, 2011 at 4:02 am | [Permalink](#)

Hi

Thanks for the tutorial. Would you mind adding the actual code files to cross check?

thanks

[Reply](#)



kerem

Posted June 6, 2011 at 8:34 am | [Permalink](#)

Hi

I cannot make this to work 😞 Here is my typed code

<http://pastebin.com/nUJxcL3q>

and here is the errors. I would appreciate any help

```
$ g++ OpenCV.c -I/usr/local/include/opencv -lhighgui -lcvaux -lxcvcore
```

```
/tmp/cc4TIHJV.o:OpenCV.c:function GetThresholdedImage(_IplImage*): error: undefined reference to `cvCvtColor'  
/tmp/cc4TIHJV.o:OpenCV.c:function main: error: undefined reference to `cvMoments'  
/tmp/cc4TIHJV.o:OpenCV.c:function main: error: undefined reference to `cvGetSpatialMoment'  
/tmp/cc4TIHJV.o:OpenCV.c:function main: error: undefined reference to `cvGetSpatialMoment'  
/tmp/cc4TIHJV.o:OpenCV.c:function main: error: undefined reference to `cvGetCentralMoment'  
collect2: ld returned 1 exit status
```

[Reply](#)



Utkarsh

Posted June 17, 2011 at 4:20 pm | [Permalink](#)

Looks like you've not included some library. Try adding -lcv to the command.

[Reply](#)



MZT

Posted June 8, 2011 at 3:55 am | [Permalink](#)

Thank you for this tutorial. It's really interesting and helped me a lot.
Do you have any idea how to know the size of the ball? Thanks in advance!!!

[Reply](#)



Utkarsh

Posted June 10, 2011 at 10:41 am | [Permalink](#)

To get the size, you could use the area of the ball detected.

[Reply](#)



DRQT

Posted June 14, 2011 at 1:53 am | [Permalink](#)

Hello I found this tutorial very interesting and really well done. thanks
I followed it and it works but I have a problem with the camera frame rate. I have a webcam video very slow (2-3 frames per second).
Is this normal?
something wrong?
you can do something to improve the frame rate?

I thank you in advance.

P.s. My camera usually works well with its default program

[Reply](#)



Utkarsh

Posted June 17, 2011 at 4:06 pm | [Permalink](#)

Maybe the code uses the camera at a very high resolution. Did you check the image sizes?

[Reply](#)



DRQT

Posted June 18, 2011 at 6:55 pm | [Permalink](#)

640×480 is too high???

[Reply](#)



Utkarsh

Posted June 18, 2011 at 7:34 pm | [Permalink](#)

It isn't. Do other applications work fine at this resolution? Or maybe you're doing a lot of processing in every frame?

[Reply](#)



DRQT

Posted June 18, 2011 at 8:36 pm | [Permalink](#)

I do not think I do too much processing: thresholding, calculation of moments and cvAdd.

I tried to do the processing on the image scaled to 320×240 and now works a bit faster.

Maybe I should try this application on a computer more powerful but camera works well with its default program (it's strange).

Thank you for your advices

[Reply](#)



Utkarsh

Posted June 19, 2011 at 12:50 am | [Permalink](#)

What hardware are you using this on?

[Reply](#)



DRQT

Posted June 20, 2011 at 3:37 am | [Permalink](#)

The hardware I am using is : Acer Intel Atom CPU N270 1,66 GHz and 1 GB Ram.

[Reply](#)



Prashad

Posted June 16, 2011 at 8:26 am | [Permalink](#)

hy utkarsh, can you help me to track multiple colored object in one frame?
i will detect hand and head with skin color detect.
what must i do first?

thx before..

[Reply](#)



Utkarsh

Posted June 17, 2011 at 4:08 pm | [Permalink](#)

You need to use [contours](#). Each blob turns into a contour that you can track separately.

[Reply](#)



Smith

Posted June 19, 2011 at 11:48 pm | [Permalink](#)

i've got an example code to detect oneblob tracking, can you help me to change this code to detect atleast two blob tracking??
i use camshift..

[Reply](#)



Utkarsh

Posted June 23, 2011 at 9:10 am | [Permalink](#)

I'm already working on something like that.

[Reply](#)



surya

Posted June 27, 2011 at 5:01 pm | [Permalink](#)

hi , i am able to work with hough transform for line detection but i am not able to access data from the output of houghlines2 function. how to access that data for other functionality?
i wanted to drive robot depending upon lines detected

[Reply](#)



Luiza

Posted July 2, 2011 at 12:59 am | [Permalink](#)

Really good tutorial! Thanks!

[Reply](#)



M.M.D. Maduranga

Posted July 11, 2011 at 3:40 pm | [Permalink](#)

i face this kind of error please help me .

Error 1 ..fatal error C1083: Cannot open include file: 'opencv2/core/core_c.h': No such file or directory
c:\opencv2.1\include\opencv\opencv.hpp

[Reply](#)



M.M.D. Maduranga

Posted July 11, 2011 at 3:41 pm | [Permalink](#)

i face this kind of error please help me.

Error fatal error C1083: Cannot open include file: 'opencv2/core/core_c.h': No such file or directory
c:\opencv2.1\include\opencv\opencv.hpp

[Reply](#)



Utkarsh

Posted July 13, 2011 at 9:17 am | [Permalink](#)

It appears your OpenCV isn't setup properly. You need to setup include paths to
c:\OpenCV2.1\include\.

[Reply](#)



Dinesh

Posted July 14, 2011 at 3:33 pm | [Permalink](#)

Hey... I want to track Black color object...So how should I proceed....? any suggestions...

[Reply](#)



Utkarsh

Posted July 21, 2011 at 8:27 pm | [Permalink](#)

For black, the V component should be less than say 50. Simple as that!

[Reply](#)



Mohan Kumar

Posted July 20, 2011 at 9:52 pm | [Permalink](#)

Nice info on opencv :).i want to interface atmega16 or any other microcontroller with a camera to identify letters and colour using opencv. can u suggest a method to do this?

[Reply](#)



Utkarsh

Posted July 21, 2011 at 8:27 pm | [Permalink](#)

Not possible. A microcontroller doesn't have enough power to take images from a camera and process them as well. You need a processor or a DSP board.

[Reply](#)



Mohan Kumar

Posted July 21, 2011 at 10:10 pm | [Permalink](#)

sorry what i actually meant was to use a computer to process the images sent by the camera on the command of a microcontroller. And the computer sends commands to the microcontroller to do some action as required from the info got from images..

[Reply](#)



jodosh

Posted August 21, 2011 at 1:25 am | [Permalink](#)

My senior design was something along those lines, the Arduino Mega 2560 is nice because it has multiple serial ports to use. I hooked up 2 cameras (<http://www.robotshop.com/4d-systems-microcam-serial-jpeg-camera-module-ttl.html>) and was able to do some stereo vision using a laptop, and arduino and those cameras.

[Reply](#)



Mohand

Posted August 5, 2011 at 7:50 pm | [Permalink](#)

hello Utkarsh ,
very nice tutorial & very helpfull .
i found just one problem ,that is : I want to calculate moments using C++ here you used C ,so any help?
sorry for my poor English 😊

[Reply](#)



Mohand

Posted August 5, 2011 at 11:32 pm | [Permalink](#)

forget it ! i found how to use them with C++ interface:

```
cv::Moments ourMoment; //moments variable
ourMoment=moments(image); //calculat all the moment of image
double moment10=moment.m10; //extract spatial moment 10
double moment01=moment.m01; //extract spatial moment 01
double area=moment.m00; //extract central moment 00
great tut anyway 😊
```

[Reply](#)



Utkarsh

Posted August 9, 2011 at 7:21 pm | [Permalink](#)

Nice work!

[Reply](#)



jodosh

Posted August 21, 2011 at 1:29 am | [Permalink](#)

any ideas on how to deal with some noise in your thresholded image? I am tracking a black object, but from time to time shadows cause problems with identifying the object.

I can make the assumption that the object that I am detecting will be the largest continuous object in the scene once it is thresholded.

[Reply](#)



jodosh

Posted August 24, 2011 at 11:57 pm | [Permalink](#)

I found a decent solution. Using

```
IplImage *img2 = cvCreateImage( cvSize(img->width+10,img->height+10), img->depth, img->nChannels );
cvCopyMakeBorder(img,img2,cvPoint(5,5),IPL_BORDER_CONSTANT,cvScalarAll(255));
cvSmooth(img2,img2,CV_MEDIAN,7,0,0);
cvSetImageROI(img2,cvRect(5, 5, img->width, img->height));
cvCopy(img2,img,NULL);
cvReleaseImage(&img2);
```

I am able to deal with any noise in groups of a few pixels. The use of the cvCopyMakeBorder is to deal with noise that is in a corner.

[Reply](#)



naveen

Posted August 30, 2011 at 2:16 pm | [Permalink](#)

Hi Utkarsh, I am following u and ur activities since September 2010. I felt you are just amazing in sharing your thoughts and experiences. Actually, I want to develop an algorithm which gives me stability in sub pixel corner detection. I have done some experiments using cvFindCornerSubPix method of openCV. But that method is also not giving me stable corners. Stable corners in the sense, I am not getting same corner values(at-least 3 decimal points) in successive images acquired by camera. I am using 5MP sensor cmos camera, which is providing me raw image data. Can you suggest me some algorithm or let me know your thoughts on this.

[Reply](#)



Utkarsh

Posted September 2, 2011 at 12:43 pm | [Permalink](#)

Whoa!

And the unstable subpixels – that's one big problem. Not just in computer vision, but in pretty much every signal processing field. The way to get rid of them is something called a kalman filter.

[Reply](#)



naveen

Posted September 16, 2011 at 10:06 am | [Permalink](#)

Thanks for the suggestion. I am trying apply kalman filter on sensor data. Do you have any idea about that filter how we can apply it image info. What is the better way of doing. Or is

there any algorithm which gives sub pixel accuracy (other than openCV method cvFindCornerSubPix).

[Reply](#)



priyanka

Posted September 3, 2011 at 8:19 pm | [Permalink](#)

hey can you help in tracking any object from the live video

[Reply](#)



Utkarsh

Posted September 3, 2011 at 10:49 pm | [Permalink](#)

This was live video! Images are captured from a camera and it gets processed!

[Reply](#)



bjorn

Posted September 5, 2011 at 2:49 pm | [Permalink](#)

Hey

I am getting undeclared identifiers for & and >. What are those?

[Reply](#)



Utkarsh

Posted September 8, 2011 at 10:44 am | [Permalink](#)

I just fixed it. I think the code will work now.

[Reply](#)



Sky

Posted September 11, 2011 at 2:32 pm | [Permalink](#)

it works well! But I have problems with noise. What can I do for denosing? Thanks...

[Reply](#)



Utkarsh

Posted November 11, 2011 at 5:24 am | [Permalink](#)

To get a smooth location, you'll have to use some math – think Kalman filters!

[Reply](#)



Deiby Ramos Avila

Posted September 17, 2011 at 1:18 pm | [Permalink](#)

Muy útil para comenzar con OpenCV, gracias por compartirlo.

[Reply](#)



rathi

Posted September 24, 2011 at 1:04 am | [Permalink](#)

hy utkarsh..m Msc. I.T. student currently working on VIRTUAL MOUSE projct...can u help me out with some reference code plsss..!!!??

nd i dd run ur tracking colord objs using opencv 2.1 with visual studio2008,bt got dis error-

error LNK2019: unresolved external symbol _cvGetCentralMoment referenced in function _main
error LNK2019: unresolved external symbol _cvGetSpatialMoment referenced in function _main
error LNK2019: unresolved external symbol _cvMoments referenced in function _main
error LNK2019: unresolved external symbol _cvCvtColor referenced in function "struct _IplImage * __cdecl GetThresholdedImage(struct _IplImage *)" (?GetThresholdedImage@@YAPAU_IplImage@@PAU1@@@Z)

plss do hlp me in fixig ds errors....

thnx

tkcr

[Reply](#)



poonam

Posted January 10, 2012 at 9:04 pm | [Permalink](#)

I have detected successfully 10 colors in a single frame in real time video. but problem came when intensity of color varies as position changes , so hsv changes.At some point color disappears totally. Can you please tell me something to solve problem, or to remove shadow and light effect ??

[Reply](#)



Utkarsh

Posted January 20, 2012 at 1:27 am | [Permalink](#)

That's a problem almost everyone has. Can you modify your code to work under such conditions? Maybe taking an 'average' of the last 10 frames?

[Reply](#)



Daniel Stefanovski

Posted January 17, 2012 at 7:42 pm | [Permalink](#)

Hi,

I'm trying to adopt your code for C++ OpenCV 2.0 Syntax for my QtGui Application.

```
cv::Mat ProcessorWidget::getTresholdImage(Mat &frame)
{
    cv::Mat hsvImage;
    hsvImage.copySize(frame);
```

```

cv::cvtColor(frame, hsvImage, CV_BGR2HSV);
cv::Mat threshedImage;
cv::threshold(frame, threshedImage, double(ui->hTSlider_Thresh->value()), double(ui->lTSlider_Max->value(
return threshedImage;
}

cv::Mat ProcessorWidget::trackColoredObject(Mat& frame)
{
    // If this is the first frame, we need to initialize it
    if(!imgScribble)
    {
        imgScribble->copySize(frame); //cvCreateImage(cvGetSize(frame), 8, 3);
    }

    cv::Mat yellowThreshedImage = getTresholdImage(frame);
    cv::Moments *moments = (cv::Moments*)malloc(sizeof(cv::Moments));
    cv::moments(yellowThreshedImage, moments);

    double moment10 = cvGetSpatialMoment(moments, 1, 0);
    double moment01 = cvGetSpatialMoment(moments, 0, 1);
    double area = cvGetCentralMoment(moments, 0, 0);

    // Holding the last and current ball positions
    static int posX = 0;
    static int posY = 0;

    int lastX = posX;
    int lastY = posY;

    posX = moment10/area;
    posY = moment01/area;

    // We want to draw a line only if its a valid position
    if(lastX>0 && lastY>0 && posX>0 && posY>0)
    {
        // Draw a yellow line from the previous point to the current point
        cv::line(imgScribble, cv::Point(posX, posY), cv::Point(lastX, lastY), cv::Scalar(0,255,255), !
    }

    cv::add(frame, imgScribble, frame);
    return frame;
}

```

The Problem is there seems to be no C++ Version of "cvGetSpatialMoment", which expects cvMoment as parameter. I use cv::Moments instead.

Do you know how to do the same with cv::Moments (OpenV 2.0+ Code)?

[Reply](#)



Daniel Stefanovski

Posted January 17, 2012 at 7:47 pm | [Permalink](#)

Oh sorry I found the answer above. Thank you anyway, great Tut!!

[Reply](#)



Jose Garcia

Posted January 23, 2012 at 5:08 am | [Permalink](#)

Thank you very much!!!!

We are doing a eyetracking and you code was absolutely useful for us!

[Reply](#)



Yeshua Padilla

Posted March 13, 2012 at 12:51 pm | [Permalink](#)

hey! I just want to congratulate you for doing this. Im just reading it. Good job. Keep doing it. 😊

[Reply](#)

[« Previous](#)

[1](#)

[2](#)

Post a Comment

Your email is *never* published nor shared. Required fields are marked *

Name *

Email *

Website

Comment

You may use these HTML tags and attributes `` `<abbr title="">` `<acronym title="">` `` `<blockquote cite="">` `<code>` `<del datetime="">` `` `<i>` `<q cite="">` `<strike>` ``

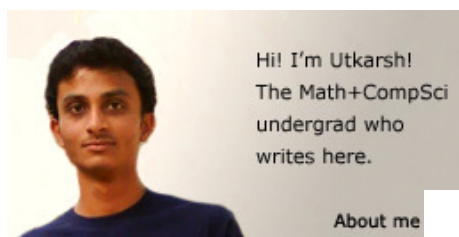
[Post Comment](#)

Free updates

Email Address

[More info](#)

1000+ subscribers!



Popular Posts

- [The Canny Edge Detector](#)
- [Implementing Canny Edges from scratch](#)

- [Image Moments](#)
- [Predator: Tracking + Learning](#)
- [Predator: Tracking + Learning](#)

More on AI SHack

Popular **Beginners** **Utkarsh's favs**

The Canny Edge Detector

Implementing Canny Edges from scratch

Image Moments

Predator: Tracking + Learning

Predator: Tracking + Learning

About me



My name is Utkarsh Sinha, and I'm an undergraduate student, pursuing B.E. Computer Science + M.Sc. Mathematics. Here, I help you understand ideas in Artificial Intelligence, using a not so techy and mathematical language. And in the process, learn more about Artificial Intelligence myself.

[Read more at the about page](#)

Created by Utkarsh Sinha - d312a67 - [Linode](#)