

```
1 #ifndef _CH_
2 #pragma package <opencv>
3 #endif
4
5 #ifndef _EiC
6 #include "cv.h"
7 #include "highgui.h"
8 #include <stdio.h>
9 #include <ctype.h>
10 #include <iostream>
11
12 using namespace std;
13
14 #endif
15 int main(){
16     CvCapture* capture=cvCaptureFromCAM(0);
17     int c;
18     IplImage* test_image = cvLoadImage("../smallOrange.jpg",1);
19     IplImage* test_hsv = cvCreateImage(cvGetSize(test_image),8,3);
20     IplImage* test_hue = cvCreateImage(cvGetSize(test_image),8,1);
21     IplImage* hist_image = cvCreateImage(cvSize(320,300),8,3);
22     CvHistogram* hist;
23     if(test_image) cout<<"source - ok"<<endl;
24     //
25
26     cvCvtColor(test_image, test_hsv, CV_BGR2HSV);
27     cvCvtPixToPlane(test_hsv, test_hue, 0, 0, 0);
28
29     //create hist
30     int size_hist = 10;
31     float hranges[] = {0, 180};
32     float* ranges = hranges;
33     hist = cvCreateHist(1, &size_hist, CV_HIST_ARRAY, &ranges, 1);
34
35     //calcalate hue` histogram
36     cvCalcHist(&test_hue, hist, 0, 0);
37
38     //draw hist on hist_test image.
39     cvZero(hist_image);
40     float max_value = 0;
41     cvGetMinMaxHistValue(hist, 0, &max_value, 0, 0);
42     int bin_w = hist_image->width/size_hist;
43
44     for(int i = 0; i < size_hist; i++ )
45     {
46         //prevent overflow
47         int val = cvRound( cvGetReal1D(hist->bins,i)*hist_image->
height/max_value);
48         CvScalar color = CV_RGB(200,0,0);
49         //hsv2rgb(i*180.f/size_hist);
50         cvRectangle( hist_image, cvPoint(i*bin_w,hist_image->height),
51             cvPoint((i+1)*bin_w,hist_image->height - val),
52             color, -1, 8, 0 );
53     }
54
55     //calcalate back projection of hue plane of input image
56     IplImage* backproject = cvCreateImage(cvGetSize(test_image), 8, 1);
57     IplImage* find_hsv = cvCreateImage(cvGetSize(test_image),8,3);
58     IplImage* find_hue = cvCreateImage(cvGetSize(test_image),8,1);
59
60     while(1){
61         IplImage* find_image =cvQueryFrame(capture); if(find_image) cout<<
"destination - ok"<<endl;
```

```
61     cvCvtColor(find_image, find_hsv, CV_BGR2HSV);
62     cvCvtPixToPlane(find_hsv, find_hue, 0, 0, 0);
63
64     cvCalcBackProject(&find_hue, backproject, hist);
65
66     cvNamedWindow("backproject", 1); cvShowImage("backproject",
backproject);
67     cvNamedWindow("src", 0); cvShowImage("src", find_image);
68     cvNamedWindow("test", 0); cvShowImage("test", test_image);
69     cvReleaseImage(&test_image);
70     cvReleaseImage(&test_hsv);
71     cvReleaseImage(&test_hue);
72
73     cvReleaseImage(&hist_image);
74
75     /*
76     cvReleaseImage(&find_image);
77     cvReleaseImage(&find_hsv);
78     cvReleaseImage(&find_hue);
79     cvReleaseImage(&backproject);
80     cvReleaseHist(&hist);
81     */
82     c=cvWaitKey(10); if(c==27)break;
83 }
84
85 cvReleaseCapture(&capture);
86 cvDestroyWindow("backproject");
87 }
88
```