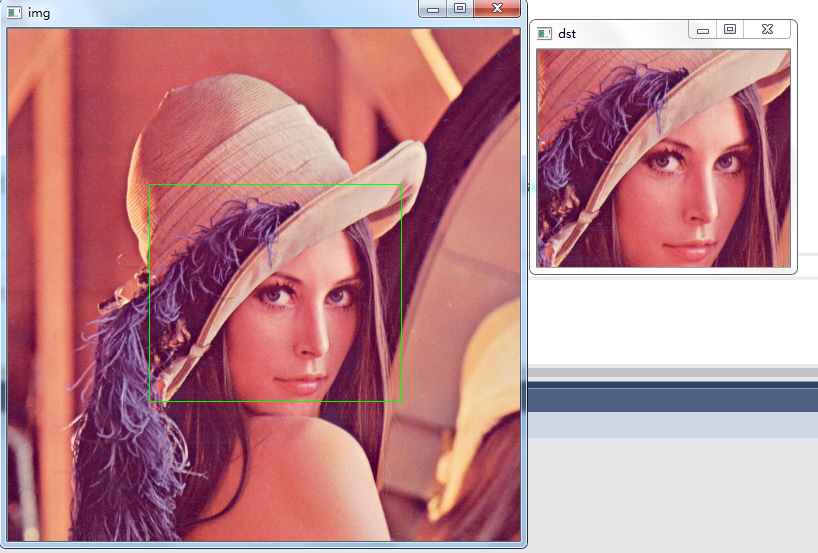
### 2.3鼠标操作图像：通过点击鼠标，选择矩形ROI区域，为后续的ROI区域处理提供方便



#include<opencv2\opencv.hpp>

#include<iostream>

using namespace std;

using namespace cv;

Mat org, dst, img, tmp;

void on\_mouse(int event, int x, int y, int flags, void \*ustc);

int main()

{

org = imread("e:\\lena.jpg");

org.copyTo(img);

namedWindow("img");

setMouseCallback("img", on\_mouse, 0);

imshow("img", img);

waitKey(0);

return 0;

}

void on\_mouse(int event, int x, int y, int flags, void \*ustc)

{

static Point pre\_pt = Point(-1, -1);

static Point cur\_pt = Point(-1, -1);

if (event == EVENT\_LBUTTONDOWN)

{

org.copyTo(img);//把org内容拷贝到img中

pre\_pt = Point(x, y);

}

else if (event == EVENT\_MOUSEMOVE && (flags & EVENT\_FLAG\_LBUTTON))//摁下左键，flags为1

{

img.copyTo(tmp);

cur\_pt = Point(x, y);

rectangle(tmp, pre\_pt, cur\_pt, Scalar(0, 255, 0, 0), 1, 8, 0);

imshow("img", tmp);

}

else if (event == EVENT\_LBUTTONUP)

{

org.copyTo(img);

cur\_pt = Point(x, y);

rectangle(img, pre\_pt, cur\_pt, Scalar(0, 255, 0, 0), 1, 8, 0);

imshow("img", img);

img.copyTo(tmp);

int width = abs(pre\_pt.x - cur\_pt.x);

int height = abs(pre\_pt.y - cur\_pt.y);

if (width == 0 || height == 0)

{

return;

}

dst = org(Rect(min(cur\_pt.x, pre\_pt.x), min(cur\_pt.y, pre\_pt.y), width, height));

imshow("dst", dst);

}

}