# 代码1 学习代码功能：

#include <iostream>

#include <opencv2\opencv.hpp>

using namespace std;

using namespace cv;

Mat addImage(Mat &s1, Mat &s2);

int main()

{

Mat src1 = imread("e:\\add1.jpg", 1);//900\*900三通道

Mat src2 = imread("e:\\add2.jpg", 1);//900\*900三通道

imshow("src1", src1);

imshow("src2", src2);

Mat dst = addImage(src1, src2);

imshow("dst", dst);

waitKey(0);

return 0;

}

Mat addImage(Mat &s1, Mat &s2)

{

Mat d(s1.size(), s1.type());

for (int i = 0; i < s1.rows; i++)

{

for (int j = 0; j < s1.cols; j++)

{

**d.at<Vec3b>(i,j)[0] = saturate\_cast<uchar>(s1.at<Vec3b>(i, j)[0] + s2.at<Vec3b>(i, j)[0]);**

**d.at<Vec3b>(i, j)[1] = saturate\_cast<uchar>(s1.at<Vec3b>(i, j)[1] + s2.at<Vec3b>(i, j)[1]);**

**d.at<Vec3b>(i, j)[2] = saturate\_cast<uchar>(s1.at<Vec3b>(i, j)[2] + s2.at<Vec3b>(i, j)[2]);**

}

}

return d;

}

# 代码2：实现图像的线性融合

#include <iostream>

#include <opencv2\opencv.hpp>

using namespace std;

using namespace cv;

int main()

{ Mat src1 = imread("e:\\add1.jpg", 1);//900\*900三通道

Mat src2 = imread("e:\\lena.jpg", 1);//512\*512三通道

resize(src2,src2,Size(src1.cols,src1.rows));//512\*512三通道---> 900\*900

imshow("src1", src1);

imshow("src2", src2);

double a,b,gamma;

cin>>a;

b=1-a;

cin>>gamma;

Mat dst; //dst =a\*src1+b\*src2+gamma; 图像 融合

addWeighted(src1,a,src2,b,gamma,dst);

imshow("dst", dst);

cout<<(int)src1.at<Vec3b>(0,0)[0]<<endl;//1

cout<<(int)src2.at<Vec3b>(0,0)[0]<<endl;//255

cout<<(int)dst.at<Vec3b>(0,0)[0]<<endl;//

waitKey(0);

system("pause");

return 0;

}

