Learn OpenCV by Examples

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

Applying Bilateral Filter

void **bilateralFilter**(InputArray src, OutputArray dst, int d, double sigmaColor, double sigmaSpace, int borderType=BORDER_DEFAULT)

Parameters:

- **src** – Source 8-bit or floating-point, 1-channel or 3-channel image.
- **dst** – Destination image of the same size and type as `src` .
- **d** – Diameter of each pixel neighborhood that is used during filtering. If it is non-positive, it is computed from `sigmaSpace` .
- **sigmaColor** – Filter sigma in the color space. A larger value of the parameter means that farther colors within the pixel neighborhood (see `sigmaSpace`) will be mixed together, resulting in larger areas of semi-equal color.
- **sigmaSpace** – Filter sigma in the coordinate space. A larger value of the parameter means that farther pixels will influence each other as long as their colors are close enough (see `sigmaColor`). When `d>0` , it specifies the neighborhood size regardless of `sigmaSpace` . Otherwise, `d` is proportional to `sigmaSpace` .

You can find a nice example in OpenCV Documentation.

A **bilateral filter** is non-linear, edge-preserving and noise-reducing smoothing filter. The intensity value at each pixel in an image is replaced by a weighted average of intensity values

SEARCH CONTENTS OF THIS BLOG

Search

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

6 Sobel Edge Detection

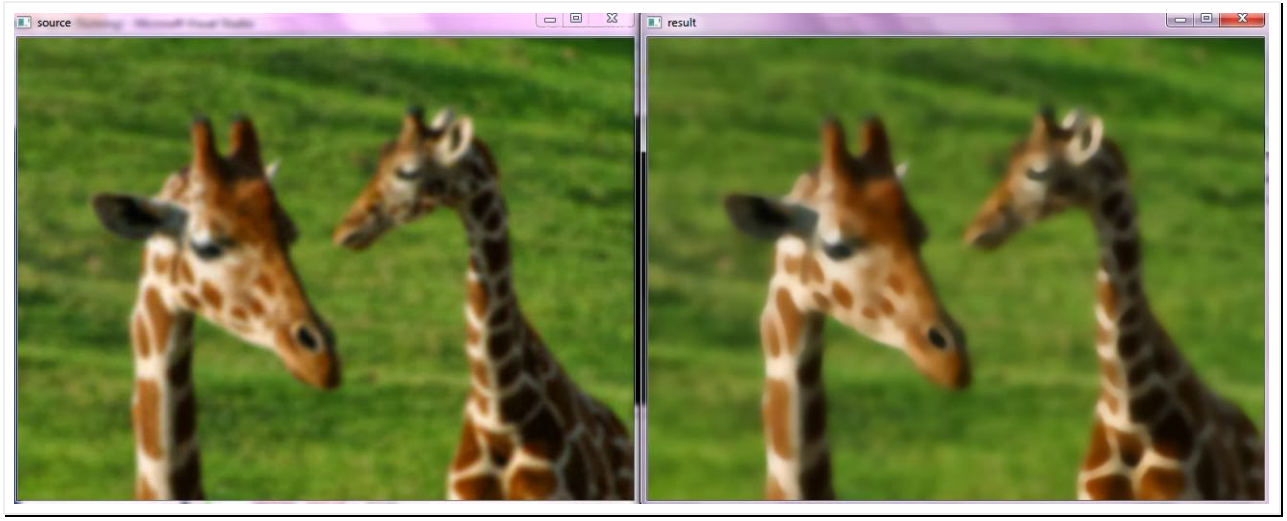
from nearby pixels. This weight can be based on a Gaussian distribution. Crucially, the weights depend not only on Euclidean distance of pixels, but also on the radiometric differences. For example, the range difference such as color intensity, depth distance, etc. This preserves sharp edges by systematically looping through each pixel and adjusting weights to the adjacent pixels accordingly.

Example:

```
#include "opencv2/imgproc/imgproc.hpp"
#include "opencv2/highgui/highgui.hpp"

using namespace std;
using namespace cv;

int main( int argc, char** argv )
 {

     Mat src = imread( "Fig1039.tif", 1 );
     Mat dst;

     //Apply bilateral filter
     bilateralFilter ( src, dst, 15, 80, 80 );
     imshow("source", src);
     imshow("result", dst);

     waitKey(0);
     return 0;
 }
```

Result:

CATEGORIES



Source:
http://en.wikipedia.org/wiki/Bilateral_filter

Labels: Filter

No comments:

Post a Comment