

阿洲的程式教學

關於Qt、OpenCV、影像處理演算法

像素強度變換(convertTo)

這邊介紹影像的強度變換，指的是對每個像素依序進行同樣的運算，假設 \mathbf{r} 和 \mathbf{s} 分別為輸入和輸出影像任一點的灰階值，可以定義為： $\mathbf{s} = \mathbf{T}(\mathbf{r})$ ，其中 \mathbf{T} 為強度變換，表示輸入和輸出圖間強度的某種映射關係，強度變換可分為線性變換和非線性變換。

最基本的線性變換就是一維線性變換： $\mathbf{x}' = \mathbf{a} * \mathbf{x} + \mathbf{b}$

- 其中 \mathbf{a} 為斜率， \mathbf{b} 為在 \mathbf{y} 軸上的截距， \mathbf{x} 表示輸入影像的灰階值， \mathbf{x}' 為輸出影像的灰階值。
- 當 $\mathbf{a} > 1$ 時，輸出圖像的對比度增加，當 $\mathbf{a} < 1$ 時，輸出圖像的對比度減小。
- 當 $\mathbf{a} = 1$ 且 $\mathbf{b} \neq 0$ 時，所有像素的灰階值增加或減少，使輸出圖像看起來更亮或更暗。
- 當 $\mathbf{a} = -1$ 且 $\mathbf{b} = 255$ 時，輸出影像的灰階值正好反轉，因為人的視覺特性的關係，這通常用來強調暗色影像中亮度較大的細節部分。

線性強度變換可以解決整體過亮或過暗的問題，但是對細節的改善有限，非線性變換才能對細節強度改善有較明顯的效果。

OpenCV 影像線性變換

`void Mat::convertTo(OutputArray m, int rtype, double alpha=1, double beta=0)`

- \mathbf{m} ：輸出圖，如果和呼叫的`Mat`尺寸或型態不同，會再重新分配空間。
- \mathbf{rtype} ：指定輸出圖型態，如果為負數的話，輸出圖型態會和呼叫的`Mat`相同。
- \mathbf{alpha} ：選擇性的放大倍率，也就是線性變換： $\mathbf{x}' = \mathbf{a} * \mathbf{x} + \mathbf{b}$ 這個式子裡的 \mathbf{a} 。
- \mathbf{beta} ：選擇性的偏移量，也就是線性變換： $\mathbf{x}' = \mathbf{a} * \mathbf{x} + \mathbf{b}$ 這個式子裡的 \mathbf{b} 。
- 函式內有呼叫`saturate_cast<>`，避免發生`overflow`的現象。

以下示範兩種方法對影像進行線性變換，分別是自己寫的`linearTrans()`和OpenCV的`convertTo()`，兩種函式會帶來相同的結果：

```
#include <cstdio>
#include <opencv2/opencv.hpp>
using namespace cv;

void linearTrans(const Mat &src, Mat &dst);

int main(){
    Mat src = imread("lena.jpg",CV_LOAD_IMAGE_UNCHANGED);
    Mat dst1;
    Mat dst2;
    linearTrans(src, dst1);
    src.convertTo(dst2,-1,1.5,30);

    imshow("window1", src);
    imshow("window2", dst1);
    imshow("window3", dst2);
    waitKey(0);

    return 0;
}

void linearTrans(const Mat &src, Mat &dst){
    dst.create(src.size(),src.type());
    int widthLimit = src.channels() * src.cols;
    for(int iH=0; iH<src.rows; iH++){
        const uchar *curPtr = src.ptr<const uchar>(iH);
        uchar *dstPtr = dst.ptr<uchar>(iH);
        for(int iW=0; iW<widthLimit; iW++){
            dstPtr[iW] = saturate_cast<uchar>(1.5*curPtr[iW]+30);
        }
    }
}
```





以下為兩種較常見的非線性強度變換，分別為對數變換和伽瑪變換：

對數變換： $x' = c \cdot \log(1 + x)$

- x' 為輸出像素值， x 為原始像素值， c 為比例常數。
- 此變換可以增強圖像中較暗部分的細節，通常在傅立葉頻譜時，強度範圍非常大，直接顯示頻譜時，顯示設備的範圍往往不能滿足，進而丟失大量暗部細節，這時可以使用對數變換，將範圍進行非線性壓縮，以至於能夠清楚的顯示影像。

伽瑪變換： $x' = (x + esp)^r$

- x' 與 x 的範圍皆為0到1， esp 為補償係數， r 為伽瑪係數。
- 與對數變換不同，伽瑪變化可以根據 r 的不同，增強低灰度或高灰度區域的對比度。
- 當 $r > 1$ 時，高灰度區域對比度增強。

- 當 $r < 1$ 時，低灰度區域對比度增強。
- 當 $r = 1$ 時，此變換為線性變換。
- 伽瑪變化並不僅可以改變影像的對比度，還能增強細節，帶來整體效果的改善。

[回到首頁](#)

[回到OpenCV教學](#)

參考資料：

[OpenCV 教程](#)

📅 2015-11-25 👤 阿宅 📁 OpenCV, 影像操作 🔑 convertTo, 像素強度變換

0 Comments 猴子遇到0與1! 程式學習筆記

1 Login ▾

♥ Recommend

🔗 Share

Sort by Best ▾



Start the discussion...

Be the first to comment.

ALSO ON 猴子遇到0與1! 程式學習筆記

Qt主窗口(Top Level Window)

1 comment • 6 months ago

mike — 喔喔

文件對話框(QFileDialog)

1 comment • 6 months ago

楊政穎 — dialog.cpp 裡面的 QString s
=
QFileDialog::getOpenFileName(this, tr

✉ Subscribe

🔗 Add Disqus to your site Add Disqus Add

🔒 Privacy

自豪的採用 WordPress

