

copyright@tingmu

3. 图像变换

图像处理的方法主要分为两大类：1. 空间域处理法 2. 频域法(变换域法) 其中最关键的预处理便是变换处理。为了有效地和快速地对图像进行处理和分析，常常需要将原图在图像空间的图像以某种形式变换(正变换)到另外一些空间，并在那里进行一些特定的变换，最后再变换回图像空间(反变换或逆变换)以得到所需要的效果。

空间域 正变换 变换域
(2-D平面) 逆变换 (频率域)

图像变换特点：对图像信息进行变换，使能量保持但重新分配。变换域中图像能量集中分布在低频成分上，边缘、线条信息反映在高频成分上。

图像变换对缓慢变化的区域对应变换后低频分量部分，对快速变化的区域对应变换后高频分量部分。

用途：1. 广泛应用于图像增强、特征提取、图像恢复、图像压缩编码等方面。

1. 提取图像特征：目标物边缘： $F(u, v)$ 高频分量。

2. 图像压缩：正变换能量集中，对集中部分进行编码。

3. 图像增强：低通滤波、平滑滤波、高通滤波、锐化滤波。

离散傅里叶变换在数字信号处理和数字图像处理中应用十分广泛，建立了离散时域和离散频域的联系。

图像是二维函数，常用二维离散傅里叶变换。图像的频率是表征图像中灰度变化剧烈程度的指标，是衡量在平面空间上的梯度。

从物理效果看，傅里叶变换将图像从空间域转换到频率域，逆变换相反。物理意义是将图像的灰度分布函数变换为图像的频率分布函数。

逆变换相反。

离散~计算量大运算时间长。快速~不是一种新的变换，它是消除离散~快速重复运算得到的，从提高了运算速度，能作实时处理。

Matlab: $f = \text{fft2}(A)$; $fc = \text{fftshift}(f)$;

% 将变换后图像频谱中，以原点和低频为中心，

$\text{imshow}(\log(\text{abs}(fc)), [J])$;

% $\log(\text{abs}(fc))$ 使用对数变换增强后的频谱，

$\text{imshow}(I, [low, high])$ 用于将非图像数据以灰度图形式显示出来。

二维离散~的性质。

① 可分离性：可分解成两个一维变换顺序执行。

② 周期性：~的幅值不变(移不变性)

$f = \text{fft2}(img1)$; $f = \text{fftshift}(f)$;

$\text{imshow}(\log(\text{abs}(f)), [J])$;

$[m, n] = \text{size}(img1)$; for $i=1:m$ for $j=1:n$

$imgnew(i, j) = ((-1)^{(i+j)}) * img1(i, j)$; end end

$f = \text{fft2}(imgnew)$; $\text{imshow}(\log(\text{abs}(f)), [J])$;

③ 周期性：用 $F(u, v)$ 反变换求 $f(x, y)$ 只需 $F(u, v)$ 中一个完整周期。

④ 共轭对称性：说明变换后幅值以原点为中心对称。在求一个周期内的值时，只需求出半个周期。

⑤ 旋转不变性：如是在空间域函数 $f(x, y)$ 域中也旋转 $-\theta$ 角。反之， $f(u, v)$ 旋转 θ 角，其反变换 $f(x, y)$ 也旋转 θ 角。

Matlab: $\text{global } img_src$;

$B = \text{imrotate}(img_src, 300, 'bilinear', 'crop')$;

$\text{figure}, \text{imshow}(B)$; $f = \text{fft2}(B)$;

$fc = \text{fftshift}(f)$; $\text{figure}, \text{imshow}(\log(\text{abs}(fc)), [J])$;

⑥ 对配性和比例性：分配性表明，~和反变换对加法可以分配，而乘法不行。比例性表明，对于二个标量 a 和 b ，有 $a f(x, y) \leftrightarrow a F(u, v)$

$f(ax, by) \leftrightarrow \frac{1}{|ab|} F(\frac{u}{a}, \frac{v}{b})$ 在空间比例尺度的展宽，相应在频域中比例尺度的压缩。幅值也减少为原来的 $\frac{1}{|ab|}$ 。

⑦ 平均值性质：若求二维离散信号 $f(x, y)$ 的平均值，只需算出相应的 $\sim F(u, v)$ 在原点的值 $F(0, 0)$ 。

⑧ 卷积定理：两个二维函数在空间域中的卷积可用求其相应的两个~乘积的逆变换而得。反之，在频域中的卷积可用空间域中乘积的~而得。

离散K-L变换(霍特林变换)

以图像的统计性质为基础：变换核矩阵由图像阵列的协方差矩阵的特征值和特征向量所决定。又称特征向量变换。

小波变换：应用：信号处理、图像处理、模式识别

小波变换是空间(时间)和频率的局部变换。

4. 图像增强

定义：按特定的需要突出图像中的某些信息，同时削弱或去除某些不需要的信息的方法。

目的：使处理后图像对某种特定应用来说，比原始图像更适用。

前提：不考虑图像降质的原因。

结果：改善后不一定逼近原图像。

① 并不能增加原始图像信息 ② 增强后好坏难以定量描述。

图像增强技术可分为两大类：1. 空间域法：主要是在空间域对图像像素灰度值直接运算处理。2. 频域法：在图像的某种变换域内，对图像的变化值进行运算，然后通过逆变换获得图像增强效果。

空间域法：点运算、灰度级处理、灰度直方图、滤波运算、平滑滤波、锐化滤波、边缘检测

频域法：低通滤波、高通滤波、同态滤波

基于点运算的增强

1. 直方图反转变换

① 图像反转：将原始图像的灰度值进行反转

Matlab: $I = \text{imread}('a.bmp')$; $J = \text{double}(I)$;

% 转换为double类型

$J = -J + (256-1)$; % 图像反转线性变换

$H = \text{uint8}(J)$; % double转uint8

$\text{imshow}(H)$;

② 灰度线性变换：灰度局限在很小范围内，进行线性拉伸

2. 灰度直方图的处理

反映灰度级与出现这种灰度的频率之间的关系。横坐标：灰度级，纵坐标：灰度级的频率。灰度直方图反映一幅数字图像像素统计特性。

直方图包含了该图像中某一灰度值的像素出现的概率，而丢失了其所在位置的像素。图像与直方图是一对一的映射关系。

Matlab: $I = \text{imread}('dog.bmp')$;

$\text{subplot}(1, 2, 1); \text{imshow}(I)$;

$\text{subplot}(1, 2, 2); \text{imhist}(I)$;

$J = \text{imadjust}(I, [0.28, 0.82], [0, 1])$;

% $[0.28, 0.82]$ 需映射 $[0, 1]$ 指定希望映射的灰度值范围。

$\text{subplot}(2, 2, 3); \text{imshow}(J)$;

$\text{subplot}(2, 2, 4); \text{imhist}(J)$;

直方图修正法包括直方图均衡化及直方图规定化两类。

(1) 直方图均衡化是将原图像通过某种变换，使原图像的灰度直方图修正为均匀分布的直方图的一种方法，是用累积分布函数作变换函数的直方图修正法。(近似的)

原理：一幅 $f(x, y)$ 灰度级分布在 $[0, r-1]$ 范围内，可以对 $[0, 1]$ 区间内的任一个值进行如下变换： $s = T(r)$ ，每个 r 都对应产生一个 s 值。 $T(r)$ 应满足：(1) 在 $0 \leq r \leq 1$ 区间内， $T(r)$ 值单调增加；(2) 对于 $0 \leq r \leq 1$ ，有 $0 \leq T(r) \leq 1$ 。

Matlab: $I = \text{imread}('boy.bmp')$;

$J = \text{histeq}(I)$ % 进行均衡化

$\text{figure}, \text{imshow}(I)$ $\text{figure}, \text{imshow}(J)$

$\text{figure}, \text{imhist}(I, 64)$ $\text{figure}, \text{imhist}(J, 64)$

(2) 直方图规定化是使原图像灰度直方图变成规定形状的直方图并对图像作修正的增强方法。

基于空域滤波的增强

空域滤波是在图像空间借助模板进行卷积操作完成的。

根据功能分成平滑滤波和锐化滤波。平滑滤波可用低通滤波实现。目的：a. 消除噪声，b. 实现模糊。

锐化滤波可用高通滤波实现。目的是为了增强被模糊的细节。

(平滑)低通滤波器：衰减或滤除高频部分，低频部分可以无衰减通过。

(锐化)高通滤波器：衰减或滤除低频部分，高频部分可以无衰减通过。

带通滤波器：衰减或滤除某一特定频率区域内的信号分量。

模板卷积运算

实现图像滤波。

卷积运算中的图像边界问题②计算结果可能超出灰度范围,超出[0,255]时,可以置0或255.

图像平滑滤波

1. 邻域平均法: 用几个像素灰度的平均值来代替每个像素的灰度.

优点: 算法简单, 计算速度快.
缺点: 图像模糊, 采用阈值法减少由邻域平均产生的模糊效应.

2. 中值滤波: 是对一个滑动窗口内的像素灰度值排序, 用中值代替窗口中心像素的原来灰度值. (消除孤立的噪声点) (由邻域像素的中间值决定窗口中心位置的像素灰度用窗口内所有像素灰度的中值代替. (二维))

注意: ①具有丰富尖顶几何结构的图像一般用十字形窗口.

②有较多点、线和尖顶的细节结构, 最好不要使用中值滤波法. poisson 泊松噪声

Matlab: $I = \text{imread}('cat.bmp');$
 $J_1 = \text{imnoise}(I, 'gaussian', 0, 0.02);$
% 添加均值为0, 方差0.02的高斯噪声.
 $J_2 = \text{imnoise}(I, 'salt \& pepper', 0.02);$
% 添加密度0.02的椒盐噪声.
 $J_3 = \text{imnoise}(I, 'speckle', 0.02);$
% 添加乘性噪声.

fspecial函数用于创建预定义的滤波算子.
 $h = \text{fspecial}(\text{type})$ average 均值滤波器.
laplacian 拉普拉斯算子.
prewitt Prewitt算子.
sobel Sobel算子.
 $h = \text{fspecial}('average', n)$ 创建n*n的均值滤波器.

$B = \text{filter2}(h, A)$ 返回图像A经算子h滤波的结果.

经邻域平均处理后, 图像的噪声得到了抑制, 但变得相对模糊. 平滑效果与采用的邻域半径(模板大小)有关.

$I_1 = \text{medfilt2}(J_1, [5, 5]);$ % 对高斯噪声图像进行5*5方形窗口中值滤波.
对有椒盐、高斯噪声的图像, 进行中值滤波. 对于消除孤立点和线段的干扰中值滤波十分有效, 对于高斯噪声则不佳.
中值滤波优点: 去噪声的同时, 还能保护边缘信息.

锐化目的: 增强图像的边缘或轮廓.

图像平滑——通过积分过程——使边缘模糊.

图像锐化——通过微分过程——边缘突出, 清晰. 梯度的绝对值与灰度差成正比.

→ 使细节清晰, 达到锐化.

拉普拉斯算子

$$\nabla^2 f = [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] - 4f(x, y)$$

$$g(x, y) = f(x, y) - \nabla^2 f(x, y)$$

Matlab: $I = \text{imread}('lena.bmp');$
 $\text{subplot}(1, 3, 1), \text{imshow}(I);$
 $h = [0, 1, 0, -4, 1, 0, 0];$ % 拉普拉斯算子
 $J = \text{conv2}(I, h, 'same');$ % 二维卷积运算
 $J = \text{uint8}(J); \text{subplot}(1, 3, 2), \text{imshow}(J);$
 $K = I - J;$ % 增强图像为原始减滤波图.
 $\text{subplot}(1, 3, 3), \text{imshow}(K);$

其他算子(Matlab)

$I = \text{imread}('lena.bmp');$ $\text{subplot}(2, 2, 1), \text{imshow}(I);$
 $\text{subplot}(2, 2, 2), \text{imshow}(I1);$ $H2 = \text{fspecial}('sobel');$
% 选择sobel算子 $I2 = \text{filter2}(H2, I);$
 $\text{subplot}(2, 2, 3), \text{imshow}(\text{uint8}(I2));$
 $H3 = \text{fspecial}('prewitt');$ $I3 = \text{filter2}(H3, I);$
 $\text{subplot}(2, 2, 4), \text{imshow}(\text{uint8}(I3));$

基于频率变换的图像增强.

目的: ①消除噪声 ②突出边缘.

用低通滤波器抑制高频成分, 可以平滑图像. 理想低通滤波器——具有振铃或模糊现象.

Matlab: $F = \text{fft2}(\text{noisy});$ % 傅里叶变换, noisy是添加噪声的图像.
 $F = \text{fftshift}(F);$ % 将频谱中心移到频谱点阵列中心.
 $[M, N] = \text{size}(F);$
% 求矩阵大小. $u1 = \text{fix}(M/2); u2 = \text{fix}(N/2);$
 $D0 = 40;$ % 截止频率.
for $u = 1:M$ for $v = 1:N$

$D = \text{sqrt}((u-u1)^2 + (v-u2)^2);$
if ($D \leq D0$) $H = 1;$ else $H = 0;$
end % 理想低通滤波器
 $\text{result}(u, v) = H * F(u, v);$ end end
 $\text{result} = \text{ifftshift}(\text{result});$
 $J1 = \text{ifft2}(\text{result});$ % 逆变换

$J2 = \text{uint8}(\text{real}(J1)); \text{imshow}(J2);$
Butterworth低通滤波器: 边缘模糊程度大大减小, 没有振铃效应产生.

频率域锐化: 边缘、细节主要位于高频成分. 锐化为了消除模糊, 突出边缘, 采用高通滤波器让高频成分通过, 低频衰减. 再傅里叶逆变换得到锐化的图像.

第五章 图像复原

图像退化: 是指图像在获取、处理、传输过程中, 由于成像系统、传输介质、设备不完善, 使图像质量变坏.

图像复原: (图像恢复): 尽可能恢复退化图像的本来面目, 沿图像退化的逆过程处理.

复原与增强的区别: 相似: 都是改善图像的质量. 不同: 增强不考虑如何退化, 采用各种技术增强视觉效果. 不考虑是否符合原有图像, 是否失真. 而图像复原要求对降质原因有一定了解, 采用相应逆处理, 恢复或重建原来图像.

复原过程: 找退化原因 → 建立退化模型 → 恢复图像 ← 反时推演

复原主要取决于对图像退化过程的了解, 知识所掌握的程度.

降质原因: 运动模糊 几何失真 散焦模糊 噪声干扰.

退化模型: 物理~ 数学~
图像复原方法: 代数~ 统计~ 频域~ 其他~
(无约束复原方法) 有约束复原方法
逆滤波 维纳滤波(最小二乘滤波) 几何畸变校正 人机交互式

Matlab实现去模糊: *

$I = \text{imread}('a.bmp');$
 $LEN = 30;$ 设置运动模糊30个像素.
 $THETA = 20;$ % 设置运动角度为20度.
 $\text{PSF} = \text{fspecial}('motion', LEN, THETA);$
% 建立二维仿真线性运动滤波器PSF.
 $\text{MF} = \text{imfilter}(I, \text{PSF}, 'circular', 'conv');$
% 用PSF产生退化图像.
 $\text{figure}(1), \text{imshow}(\text{uint8}(\text{MF}));$
 $\text{wnr1} = \text{deconvwnr}(\text{MF}, \text{PSF});$
% 用Wiener滤波消除运动模糊图像.
 $\text{figure}(2), \text{imshow}(\text{uint8}(\text{wnr1}));$
deconvwnr 强模糊图像复原函数.
wiener2 强模糊图像空间域锐化的作用, 只支持二维滤波灰度图.

第4章 图像分割

图像分割是把图像分成若干个有意义的区域的处理技术。是指根据灰度、颜色、空间纹理、几何形状等特征把图像划分成若干个互不相交的区域，使其在同一区域内表现出一致性或相似性，而在不同区域间明显不同。就是在一幅图像中，把目标从背景中分离出来，以便进一步处理。

图像分析系统的基本构成：

获取图像 → 预处理 → 图像分割 → 图像识别 ← 特征提取

图像分割是图像识别的前提。图像分割质量的好坏直接影响后续图像处理的效果。

分类：

基于阈值的分割

直方图阈值分割法
类间阈值
二值最大熵
模糊

基于边缘的分割
点检测
线检测
边缘检测

区域
区域增长
区域分裂合并

运动的检测
直接方法
间接方法

基于阈值的分割

选择一个合适的阈值，以确定每个像素点是属于目标，还是背景区域，从而产生相应的二值图像。

另外，可设定一个灰度范围。

特殊情况下，高于阈值的像素保持原灰度，其它为0。

阈值选取是关键，过高，则过多目标点被误判为背景，过低则相反。

简单直方图分割法：画面比较简单且对景物灰度分布比较有规律时，背景和对景物在灰度直方图上各自形成一个波峰，低谷处即为阈值，可得2区域分割。

Matlab: `imgnew = im2bw(img1, 130/255)` %直方图
阈值分割图像 %图像二值化，根据130/255
确定阈值分为目标和背景，img1灰度图像

`figure; imshow(imgnew); title('分割后图像');`

最佳阈值：指图像中目标物与背景的分割错误最小的阈值。

类间最大熵法又称最大类间方差法，由Otsu提出，利用灰度直方图，以目标均值和背景均值之间的方差最大为依据，动态地确定分割阈值。

Matlab: `img1 = imread('rice.bmp');`
`level = graythresh(img1);`
`imgbw = im2bw(img1, level);`
`imshow(imgbw);`
`title('使用最大类间方差法获得阈值');`

基于边缘的分割

所谓边缘是图像局部特性的不连续性。灰度或结构等信息的突变处称为边缘。

边缘分类：阶跃状 脉冲状 屋顶状。

1. 可用一阶导数的幅度值来检测边缘存在。

2. 可用一阶导数的过0点检测边缘位置，用二阶导数在过0点附近的符号确定边缘像素在图像边缘的暗区或明区。

3. 通过检测脉冲状边缘的2个一阶导数过0点，就可确定脉冲中范围。

4. 通过检测屋顶状边缘剖面的一阶导数过0点，可以确定屋顶位置。

边缘检测实质：采用某种算法来提取图像中图像与背景的分界线。

边缘检测算子

可用一阶、二阶局部微分算子来检测图像中的边缘。

Marr算子 - Canny算子

Roberts算子：对噪声敏感，但比梯度算子略好。

Prewitt算子：检测边缘的同时，抑制噪声的影响。

Sobel算子：对4邻域采用加权方法计算差分，进一步抑制噪声，但检测边缘较宽。

拉普拉斯算子分析：

优点：各向同性，线性和位移不变的；对细线和孤立点检测效果较好；

缺点：①对噪声敏感 ②不能测出边缘的方向 ③产生双像素的边缘。

主要用于已知边缘像素，确定该像素是在图像的暗区域或明区边。

Matlab:

`Y = edge(img1, 'sobel', 0.1);`

%1.7sobel算子边缘检测，阈值0.1。

`figure; imshow(Y); title('sobel边缘');`

Canny算子提取边缘较完整，连续性很好，效果优于其它算子。

基于区域的分割

基本思想：将具有相似性质的像素集合起来构成区域。

步骤：①找种子像素 ②生长或相似性 ③合并区域。

生长规则：如果像素与种子像素灰度值差小于阈值T，则包括进种子像素所在的区域。

区域生长法3个步骤：

- ①选择或确定种子像素。
- ②确定在生长过程中包括进来的像素。
- ③制定生长过程停止的条件或规则。

区域分裂-合并：

先将图像分割成一系列不相交区域，然后对这些区域进行合并或分裂得到单个区域。

常用：四叉树分解法。

利用四叉树对表达方式的简单分裂-合并算法。

R: 整个区域 P: 递归调用，把R四等分R_i，并始终P(R_i)=TRUE，即：如果P(R)=FALSE，

则将其四等分，P(R_i)=FALSE，再四等分，直至R_i单个像素。

合并规则：只要两相邻区域R_i, R_j具有一致性，则可将R_i, R_j合并。

步骤：(1)对任一区域R_i，如果P(R_i)=FALSE，四等分。

(2)对相邻R_i, R_j (可大小不同，即不在同一层)，如果P(R_iUR_j)=TRUE，则合并。

(3)如果进一步分裂合并都不可能，则结束。

图像分割常用数据集：

Pascal VOC; MS COCO; CityScapes; CamVid

copyright@tingmu