



3.1 进一步的思考

第 2 章对于噪声处理算法的介绍是必要的。然而,第 2 章的研究局限于图像去噪,可以认为是一种对图像进行复原的技术。在实际工程应用中,人脸经常是运动的,这就要求将图像去噪和运动目标检测结合起来。在此基础上再结合第 1 章介绍的视频图像序列转换技术,就可以实现视频去噪。作为遮挡人脸识别的预处理技术,也需要一种非局部的滤波去噪算法,以适应人脸的运动状态。

在日常生活中,噪声是阻碍人们感觉器官理解接收源信息的因素,图像中也存在各种各样的因素阻碍人们接收图像信息,也就是所谓的图像噪声。噪声是影响数字图像质量的一个非常重要的因素,它主要来自图像的获取及传输的过程。在图像的采集过程中,摄像头或者成像传感器的性能会受到外界环境条件、传感器元件质量等诸多因素的影响。图像噪声会影响图像中表达信息的接收。除此之外,图像噪声也会影响各种图像识别处理的科学研究和各类应用,例如,人脸识别、文字识别、花卉识别、目标识别、无人驾驶技术等各种基于计算机视觉的研究与应用。

图像去噪是一种对图像进行复原的技术,是图像处理领域的典型问题。到目前为止,科学家们已经提出了很多种不同类型的去噪方法,例如,建立在概率论、统计理论、偏微分方程、线性和非线性滤波、频谱分析、多分辨率分析基础上的各种方法。本章主要研究的是图像噪声对帧间差分法进行运动目标检测的影响,以及运用图像去噪并对其改进优化。

3.2 从局部到非局部

非局部滤波的去噪效果优异,但是想要将其应用到运动目标检测还需要解决算法复杂,处理图像耗时较长的问题。对一个视频进行运动目标检测要处理几百帧图像,不可能运用传统的非均值滤波,这样太耗时而且根本不实用,因而这部分提出一种基于积分图加速原理

的快速非均值滤波算法。本节分别分析传统非局部滤波算法的原理、非局部均值滤波算法的不足、快速非局部均值滤波算法原理以及运动目标检测方法。

3.2.1 非局部滤波算法原理

其实图像中的像素点之间都是有关联的,它们都不是孤立存在的,每个像素点与其周围的像素点一起组成了图像中的几何结构。考虑到图像中复杂的空间相互关系,以像素为中心的窗口邻域(即图像块)能够很好地反映像素点的结构特征。每个像素点所对照的一组图像块的集合可以作为图像的完全表示。总体来说,图像是具有自相似性质的,即图像中不同位置的像素通常有很强的相关性,就像纹理图像一样。在图像中任意取出一个小窗口,在自然图像中包含的多种重复结构或者说冗余的额外信息,都能够在这幅图像中找到许多类似的窗口结构。足够多的重复结构也同时包含在自然图像中,例如图像平坦区域中有大量相似的像素点,在同一直线或者曲线边界上的像素点也有相似的邻域模式。对于图像中空间位置距离较近的窗口,这个结论也明显成立,而这就是局部规则性的假设。如果使用能够描述图像结构特征的图像切片来度量像素之间的相似度,将比单个像素的测量更为准确,因而可以更好地对图像的结构信息进行保护。Efros 和 Leung 是第一个注意到图像具有这个特性的,他们利用图像之间的相似性进行纹理合成,并对图像中的小孔进行填充。该算法寻找与图像中要处理的像素点相似的像素点。Buades 等在 2005 年提出了非局部均值滤波去噪算法。第一次引入了非局部滤波的概念,采用基于变换域滤波的去噪方法和基于结构相似性的局部平滑方法去除噪声,其目的主要是为了恢复图像的主要几何结构。这些方法都建立在对原始图像进行规则性假设的基础之上,由于噪声与图像的细微结构和细节信息同时具有极其相似的特征,因此平滑地定义图像的细微结构和细节信息,才能够有效地保护图像的结构信息,达成目前最优异的去噪效果。

采用非局部均值滤波图像去噪算法实现滤波去噪是对传统的邻域滤波方法的一个重大的改进。首先,该算法考虑图像的自相似性,突破了邻域滤波只用于局域滤波的限制。由于在空间位置上相似的像素点的距离并不一定会离得很近(例如,具有周期性质的图像),因此在更大的范围内找到相似的像素会更有利。其次,该算法将相似像素点定义为具有相同邻域模式的像素点,与只利用单个像素点本身信息获得的相似性相比,利用像素点周围大小固定窗口中的信息来表示像素的特征更加可靠和稳定。

对于一个给定的像素点 i 来说,图像块 $N(i)$ 是以像素点 i 为中心的大小为 $n \times n$ 的区域, $N(j)$ 是 $N(i)$ 邻域内的图像块,使用图像块 $N(i)$ 与图像块 $N(j)$ 之间的高斯加权欧氏距离来计算像素点 i 与像素点 j 之间的相似性。图像块 $N(j)$ 与图像块 $N(i)$ 之间的距离越小,说明像素点 j 与像素点 i 会越相似,累加恢复时像素点 j 赋予的权值也越大。假定滤波后图像为 $\hat{f}(i)$,噪声图像为 $f = \{f(i) | i \in \Omega\}$, Ω 是图像区域, $f(i)$ 表示像素 i 的灰度值,

则非局部均值滤波算法的具体计算如下：

$$\hat{f}(i) = \frac{\sum_{j \in I} W(i, j) f(j)}{\sum_{j \in I} W(i, j)} \quad (3-1)$$

$$W(i, j) = e^{-\frac{d(i, j)}{\sigma^2}} \quad (3-2)$$

$$d(i, j) = \| N(i) - N(j) \|_{2, \alpha}^2 \quad (3-3)$$

其中, α 为高斯核函数的标准差, 使用高斯核对图像块进行卷积处理, 能够降低噪声对距离计算的影响并突出图像块中心在像素的作用; $d(i, j)$ 表示两图像块之间的加权欧氏距离; σ 为控制平滑程度的滤波参数; I 表示以像素 i 为中心的搜索邻域, 理论上 I 应为整个图像空间, 即 $I = \Omega$, 但这种取值方法将使算法复杂度太高, 因此, 通常会把搜索窗 I 减小至一定大小; $W(i, j)$ 为加权平均时像素 j 对应的权重。

从非局部均值滤波算法的计算过程可以看出, 如图 3.1 所示的像素点 q_1 和 q_2 会获得比较大的权重, 因为像素点 q_1 和像素点 q_2 所在的图像块与像素点 p 所在的图像块更加相似; 相反, 像素点 q_3 所对应的权重 $W(p, q_3)$ 比较小, 因为像素点 q_3 所在的图像块无论是从灰度分布还是从几何结构上都与像素点 p 所在的图像块差别较大。与传统邻域平均方法相比, 非局部均值滤波算法同时结合了几何结构分配邻域像素权重值和灰度分布, 更加有效地区分不同相似度的贡献, 所以该算法能够取得更显著的去噪效果。



图 3.1 非局部均值滤波原理

3.2.2 存在的不足

非局部均值滤波算法虽然原理简单、对图像去噪的性能优越, 特别是其非局部的思想,

在图像处理邻域有着非常重要的影响,但非局部均值滤波算法也存在一些不足,主要表现在以下几方面。

(1) 相似性度量计算方法缺乏稳定性。根据上述算法原理进行相似性计算时,只有那些可以被平移到不同位置的图像块才能获得比较大的权重。实际上,如果把图像中的某些图像块旋转一定的角度,它们仍然与给定的图像块相似。如果仅使用算法中的相似性度量计算方法就有可能忽略掉这些非常相似的像素点或者图像块。

(2) 在进行相似性计算的时候,使用了高斯加权的欧氏距离能够降低噪声对距离计算的影响,并且突出图像块中心像素对相似度计算的贡献。该算法以及许多基于该算法改良的算法都是采用了这一度量方法来计算欧式距离。然而由于高斯核是各向非异性的,因此对图像块内不相似的像素点也将会采用同样的比例来参与欧式距离的计算,从而对中心像素的相似性比较带来干扰。因此,探讨如何选取适合的邻域用于欧式距离的计算,这是非常必要的。

(3) 该算法在去噪时所采取的计算策略是将搜寻邻域内的一切像素点依据不同的权重累加求平均,但搜索邻域内还包括不相似的像素点,这些不相似像素点也会参与贡献权重,这是非常不科学的。因此,如何获得更多可靠的具有相似性的像素点,从而有效地消除不同像素对去噪性能的影响将是一个值得钻研的问题。

(4) 虽然该算法原理容易理解,但是运算复杂度比较高,这也是算法的一个不足。相似度权值计算会耗费非常久的时间,所以该算法在实际工程应用中必然有局限性。

(5) 权值计算中参数的取值也会影响该算法的去噪效果。这个取值一般是经验值,没有统一的取值方式,参数选取的好坏会直接影响去噪效果。一般在滤波算法进程中会将权重参数设置为固定值,并且该值的大小仅与噪声的方差关联。但是,权重的相关参数的影响因素有两个主要的方面,一个是图像内容,另一个是噪声水平。只由噪声水平确定的固定权重参数必定没有办法同时兼顾图像中所有内容存在的差异,因而,设计一个自动适应而且还能同时兼顾噪声水平和图像内容的权重参数将会是提高该算法去噪水平所需解决的重要问题。

3.2.3 算法改进思路

因为非局部均值算法需要对图像中所有像素点灰度值做重新估计,假设图像共 N 个像素点,搜索窗大小定义为 D ,邻域窗口大小定义为 d ,计算每个矩形邻域间相似度的时间为 $O(d^2)$,对于每个像素点要计算搜索窗内 D^2 个像素点的相似度,故非局部均值滤波算法的复杂度为 $O(ND^2d^2)$ 。

使用传统的非局部均值滤波算法处理一幅 256×256 的图像,要耗费一个小时左右,这限制了该算法在实际生活中的应用,所以对非局部均值滤波算法进行加速非常必要。本节介绍采用基于积分图像的快速非局部均值算法。

积分图像加速的原理是,将图像中的全部像素点,一次性计算出某一偏离坐标点方向的

权重值,而不是计算某一点的所有偏离点。准确地,计算当前图像与一定偏移后的图像的距离并求其平方值,继而计算与原图像同等尺寸的积分图,则通过这个积分图就能够很简单地计算出每个点与偏离它一定偏移的坐标点的距离。积分图的作用体现在使用线性计算减少重复的计算,即用存储换时间。对图像整体处理,原图像与平移 $t = y - x \in [-D_s, D_s]$ 后的图像的欧氏距离为:

$$S_t = \|v(x) - v(x+t)\|_2^2 \quad (3-4)$$

先假设一个关于像素差值的积分图像:

$$S_{d_t}(x) = \sum_{\{z=(z_1, z_2) \in N^2, 0 \leq z_1 \leq x_1, 0 \leq z_2 \leq x_2\}} s_t(z), \quad x = (x_1, x_2) \in \Omega \quad (3-5)$$

式(3-5)在实际操作中可表达为:

$$\forall x = (x_1, x_2) \in \Omega, \quad x_1 \geq 1, x_2 \geq 1 \quad (3-6)$$

$$S_{d_t}(x) = s_t(x) + S_{d_t}(x_1 - 1, x_2) + S_{d_t}(x_1, x_2 - 1) - S_{d_t}(x_1 - 1, x_2 - 1) \quad (3-7)$$

那么对于不同区域的欧式距离可以写为:

$$\begin{aligned} \|v(x) - v(y)\|_{2,d}^2 = & \frac{1}{d^2} [S_{d_t}(x_1 + d_s, x_2 + d_s) + S_{d_t}(x_1 -, x_2 - d_s) - \\ & S_{d_t}(x_1 + d_s, x_2 - d_s) - S_{d_t}(x_1 - d_s, x_2 + d_s)] \quad (3-8) \end{aligned}$$

此时对于 N 个像素点的图像,搜索窗大小为 D , 计算非局部均值滤波算法的复杂度为 $O(ND^2)$ 。

明显可以发现式(3-8)与式(3-7)思想一致,如图 3.2 所示,加 2 个减 2 个并以差值大小判断偏差大小。

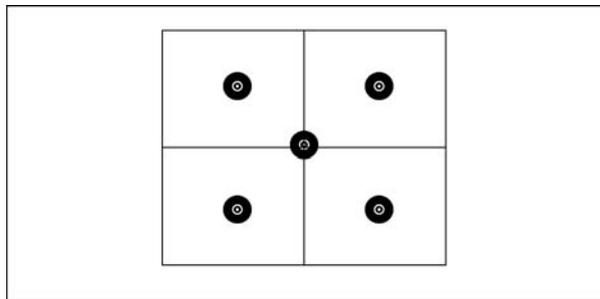


图 3.2 加 2 个减 2 个并以差值大小判断偏差大小

3.3 应用探索和尝试

3.3.1 运动目标检测

运动目标检测方法已经发展了多年,迄今为止该领域已经有了非常多的方法,但是许多

方法都是从最传统的3种方法改进而来的。

(1) 光流法用于描述相对于观察者的运动引起的观察目标表面或边缘的运动,光流是表达图像运动的有效而简单的方法,光流是图像序列中图像亮度模式的性能。因为光流场所具有的不连续性,所以可以用光流法将所要研究的图像分割成对应的不同的运动物体区域。基于光流法的运动目标检测具有运动目标随时间变化的光流特性,光流中会同时包含被观察物体的相关结构信息和运动物体的运动信息,但是在实际应用中,由于遮挡、噪声、透明度等多种因素的影响,无法更好地满足光流场基本方程假设的灰度守恒条件,从而没有办法得到正解。此外,大多数光流法计算复杂,仅仅只能得到稀疏的光流场,因而不适合实时处理。该方法并不适用于具有时效性和精确性要求的系统。

(2) 背景差分法是运动目标检测技术中常用的一种方法。该方法主要利用当前帧图像与背景图像的差值图像检测运动区域。由于该方法不受运动目标速度的限制,可以提供完整的特征数据,所以可以更完整地提取运动目标。但是,其检测性能与背景图像提取的质量有很大关系,并且容易受光照条件和外部条件引起的场景变化的影响。

(3) 帧间差分法的原理是减少图像前后两个帧所对应的像素值。当环境亮度变化不大时,如果相应的像素值相差很大,则认为场景中该像素点是运动物体;如果像素值相差小,则认为该像素点是背景。对于较大的、颜色一致的运动对象,该方法可能会在对象内部产生空洞,从而无法完全提取运动对象。当对差分图像进行二值化处理后,可能还会有很多没有使用的噪声斑点,另外目标运动物体内部也可能产生噪点,所以需要通过数学和形态学的方法,对差分后的二值图像进行滤波处理。该方法的缺点是对环境的噪声特别敏感,所以对二值化时阈值的选择有很高的要求,如果选择太低就不足以抑制图像中的噪点,选择过高则忽略了图像中有用的变化。

根据帧间差分法原理,当视频中的确有物体在运动时,相邻的两帧或者相邻的三帧之间的图像在灰度上会有所差别,只要求取两帧灰度化图像矩阵差的绝对值,则静止物体在差分图像矩阵上表现出来全是0,而视频中的运动物体(尤其是运动物体的边缘轮廓处)由于存在灰度变化包含不为0的点,当所求差值绝对值超过设定阈值时,就可以将此物体判断为所求的运动目标,从而简单实现了目标的检测功能。

帧间差分法算法简单且易于实现,程序设计的复杂度不高,同时其对场景的光线变化不是特别敏感,还能够适应各种各样的动态环境,有比较强的鲁棒性。但其缺点是受各种不确定因素的影响,比如光线的变化、部分背景物体小幅度晃动等,这些因素会使视频图像产生大量的噪声,帧间差分法原理图如图3.3所示。

综上所述,传统的帧间差分法会受各种不确定因素的影响,使得视频图像会产生大量的噪声,从而导致分割后的二值图像有很多噪声斑点,因此本章结合非局部均值滤波对差分后二值图像进行滤波后再采用帧间差值法进行运动目标检测,减少背景物体小幅度晃动以及大量噪声的干扰,从而改进优化帧间差分法。改进后的原理如图3.4所示。

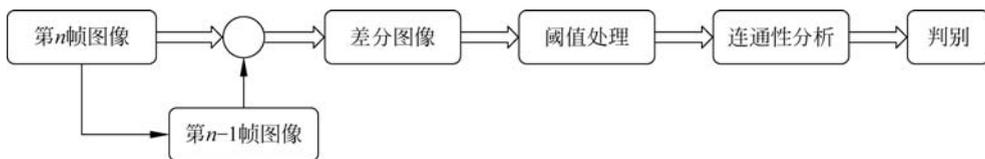


图 3.3 帧间差分法原理图

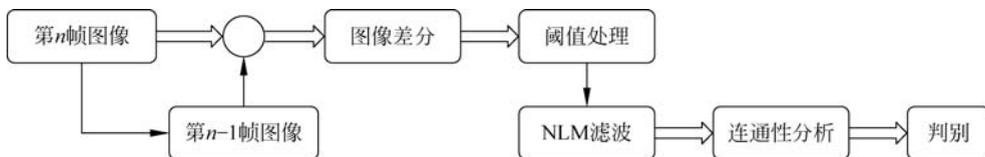


图 3.4 结合了非局部均值滤波的帧间差分法原理图

3.3.2 算法设计思想

本设计的程序包含非局部均值滤波、背景差分与帧间差分结合以及非局部均值滤波的积分图快速算法实现,程序要实现将视频文件分解为一帧一帧的图像,并对每一张图像进行背景差分和帧间差分后再进行非局部均值滤波处理,最后进行运动目标检测。

为了观察加入非局部均值滤波后的帧间差分法检测运动物体是否去除了噪声影响,因此设计了一个有对比的实验,分别要对未滤波的差分二值图像和已滤波的差分二值图像进行运动目标检测。此次实验和编程环境都是基于 Windows 操作系统以及 MATLAB 2018b 版本。所设计的程序要实现将视频帧间差分得到每两帧的差分图像,然后将差分图像二值化并进行非局部均值滤波,滤波后找出该图像的八连通区域,则认为已经将运动目标标记出来。为方便实验,并考虑到计算机硬件条件,实验数据为 5s 的画质清晰的短视频,视频中背景静止且有物体移动但避免大量杂物移动。

本次实验的数据均为作者拍摄的背景静止但前景有物体移动的视频,分别拍摄了十几个不同的视频。每个视频的时长都是 4~7s,视频中都有不同物体或者是戴口罩的人脸的移动,选择视频背景中比较简单的进行实验。为了运行顺畅,首先用视频编辑软件压缩视频,同时尽可能保留视频画质。

3.3.3 算法设计过程

按照设计目的, MATLAB 程序需要实现以下功能。

- (1) 读取视频文件。
- (2) 读取视频的每一帧。

- (3) 确定背景。
- (4) 将每一帧与背景差分得到差分图像。
- (5) 差分图像进行二值化。
- (6) 二值化阈值自适应。
- (7) 二值化图像要进行非局部均值滤波。
- (8) 进行连通性分析,找出二值化图像的八连通区域。
- (9) 进行连通性分析,找出滤波后的图像的八连通区域。
- (10) 显示图像观察。
- (11) 保存图像序列。

其中第 3~11 个功能应该在一个循环体中。第 7 个功能是算法核心,非局部均值滤波算法设计要考虑处理视频的每一帧,处理速度要快,本设计采用了 3.1.3 节介绍的快速非局部均值滤波算法。按照以上的结构设计要求的 MATLAB 程序如下所述。

读取视频文件,文件地址 pwd 为当前工作目录,当前运行程序所保存的文件夹,文件名为“30”,格式为“MP4”如果是其他文件格式可将此直接改为所读视频文件的格式。代码如下:

```
mov = VideoReader(fullfile(pwd, '30.mp4'));
```

读取视频后将获取视频帧个数,决定程序循环次数。代码如下:

```
nframes = mov.NumberOfFrames;
```

读取第一帧图像并进行灰度化(rgb2gray)再转化为 double 类型后,以此图像作为初始背景,然后开始进入循环,直到视频最后一帧结束,其中 nStar 设为 1。代码如下:

```
Background = double(rgb2gray(read(mov, nStar)));
for k = nStar + 1 : nframes;
```

进入循环后,将当前帧(第 k 帧)与前一帧(第 $k-1$ 帧)做差分,两帧图像矩阵做差分时元素小于零取绝对值,为了防止过大跨度图像的矩阵边缘像素被擦除,要对差分后的图像进行归一化,再将图像转化为 uint8 类型。代码如下:

```
CurrentImage = double(rgb2gray(read(mov, k)));           % 当前帧
FormerImage = double(rgb2gray(read(mov, k-1)));         % 前一帧
difgrayFrame = im2uint8(mat2gray(abs(CurrentImage - FormerImage))); % 帧间差分,取绝对值,进行归一化,转化为 uint8 类型
```

将差分后的灰度图像进行二值处理,二值图像是指只有纯白(255)或者纯黑(0)的图像,二值化阈值 th 通过函数自适应。代码如下:

```
th = get_iter_th(difgrayFrame);
```



```

Ds = 5; % 搜索窗口半径
h = 10; % 滤波参数
offset = ds + Ds;
PaddedImg = padarray(src, [ds + Ds, ds + Ds], 'symmetric', 'both');
% 非局部均值滤波去噪
% 生成 3 个 m * n 全为零的矩阵
sumimage = zeros(m, n);
sumweight = zeros(m, n);
maxweight = zeros(m, n);
image = PaddedImg(1 + Ds:Ds + m + ds, 1 + Ds:Ds + n + ds);
[M, N] = size(image);
for r = -Ds:Ds
    for s = -Ds:Ds
        if (r == 0 && s == 0)
            continue; % 如果 r 等于 0, s 也等于 0, 则跳过当前点偏移
        end
        wimage = PaddedImg(1 + Ds + r:Ds + m + ds + r, 1 + Ds + s:Ds + n + ds + s); % 求得差值积分图
        diff = image - wimage;
        diff = diff.^2;
        J = cumsum(diff, 1);
        J = cumsum(J, 2);
        distance = J(M - m + 1:M, N - n + 1:N) + J(1:m, 1:n) - J(M - m + 1:M, 1:n) - J(1:m, N - n + 1:N);
        % 计算距离
        distance = distance / ((2 * ds + 1).^2);
        weight = exp(-distance ./ (h * h)); % 计算权重并获得单个偏移下的加权图像
        sumimage = sumimage + weight .* wimage(ds + 1:ds + m, ds + 1:ds + n);
        sumweight = sumweight + weight;
        maxweight = max(maxweight, weight);
    end
end
sumimage = sumimage + maxweight .* image(ds + 1:ds + m, ds + 1:ds + n);
sumweight = sumweight + maxweight;
dst = sumimage ./ sumweight;

```

进行非局部均值滤波后,分别得到了未滤波的差分二值图像和滤波的差分二值像,接下来对两张图像分别进行连通性分析,找出图像中所有八连通区域视为运动目标所在位置并且用红框标记。连通性分析的核心程序如下:

```

[L, nm] = bwlabel(CuterBW, 8);
for i = 1:nm
    [r, c] = find(L == i);
    left = min(c);
    right = max(c);
    top = min(r);
    bottom = max(r);

```

```

width = right - left + 1;
height = bottom - top + 1;
rectangle('position',[left,top,width,height],'EdgeColor','r');
end

```

将差分图像、滤波后的差分二值图像、未滤波差分后的二值图像的目标标记图像以及滤波后的差分二值图像的目标标记图像放在一个窗口中显示。实现代码如下：

```

figure(1);
subplot(1,4,1), imshow(CurrentImage, []), title(['第' num2str(k) '帧']);
[L, nm] = bwlabel(CuterBW, 8);
for i = 1:nm
[r, c] = find(L == i);
left = min(c);
right = max(c);
top = min(r);
bottom = max(r);
width = right - left + 1;
height = bottom - top + 1;
rectangle('position',[left,top,width,height],'EdgeColor','r');
end
subplot(1,4,2), imshow(Cut, []), title('差分后的图像');
subplot(1,4,3), imshow(dst, []), title('滤波后的差分二值图像');
subplot(1,4,4), imshow(CurrentImage, []), title('滤波后目标跟踪');
[L, nm] = bwlabel(dst, 8);
for i = 1:nm
[r, c] = find(L == i);
left = min(c);
right = max(c);
top = min(r);
bottom = max(r);
width = right - left + 1;
height = bottom - top + 1;
rectangle('position',[left,top,width,height],'EdgeColor','r');

```

设计所需设备为一台能正常运行 MATLAB 的计算机 (MATLAB 版本为 2018b)。实验视频保存在与 MATLAB 程序同一文件夹中。将程序语句“mov = VideoReader(fullfile(pwd, '30. mp4 '));”单引号中内容改为导入视频的文件名以及后缀。此时运行该 MATLAB 程序, 将会弹出图像显示窗口, 分别显示差分图像、滤波后的差分二值图像、未滤波差分后的二值图像的目标标记图像以及滤波后的差分二值图像的目标标记图像, 可以直接在窗口上观察实验结果。

实验流程图如图 3.5 所示。

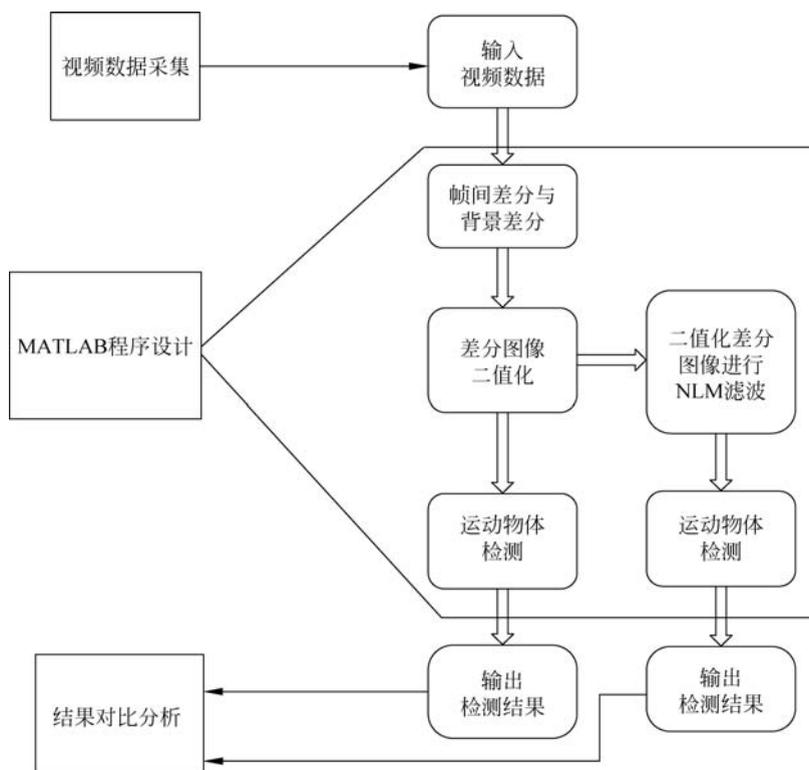


图 3.5 实验流程图

3.4 实验结果与结论

3.4.1 预测实验结果

实验结果应该是未进行滤波的目标检测出现多个方框以及每一帧由微小的变化都会有方框框出,而滤波后的目标检测应该只是视频中主要运动部位有方框框出,忽略了大部分微小的方框。

3.4.2 观察收集实验结果

实验结果用录屏软件进行录制,并且每一帧变化的图像序列都保存在计算机中,此功能可以直接在程序中加入下述语句来实现:

```

"filepath = pwd;
cd('D:\shuju\1');

```

```
saveas(1,['11/4iñ' num2str(k) '.jpg']);  
cd(filepath);"
```

保存的图像序列部分如图 3.6~图 3.14 所示。其中,图 3.6~图 3.8 所示为视频 1 的第 2~4 帧,图 3.9~图 3.14 所示为视频 1 的第 42~47 帧。



图 3.6 第 2 帧

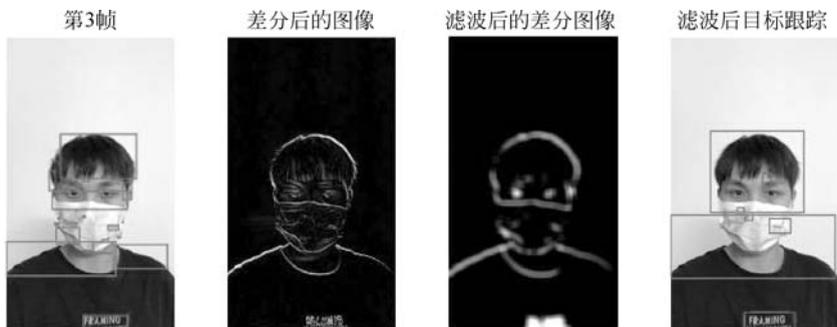


图 3.7 第 3 帧

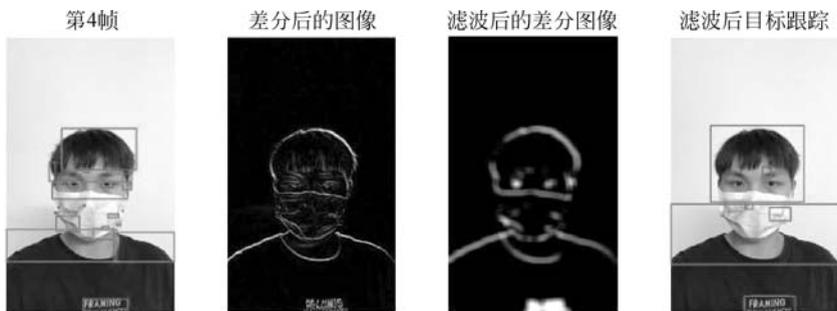


图 3.8 第 4 帧



图 3.9 第 42 帧

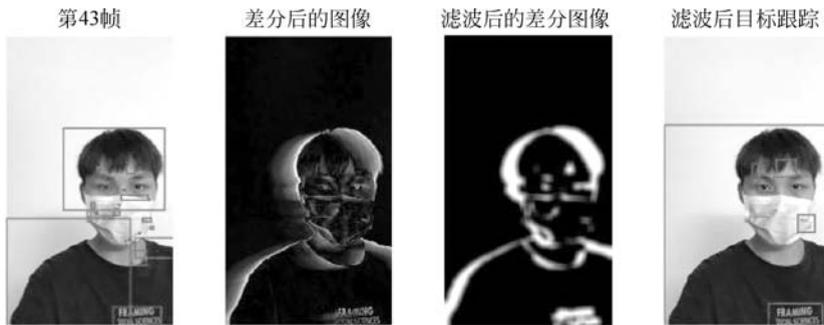


图 3.10 第 43 帧

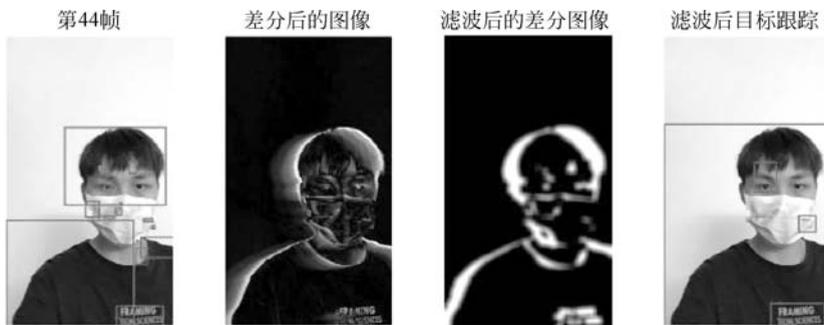


图 3.11 第 44 帧

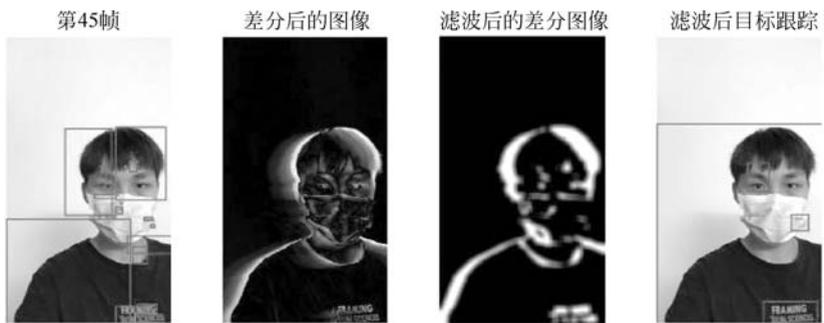


图 3.12 第 45 帧

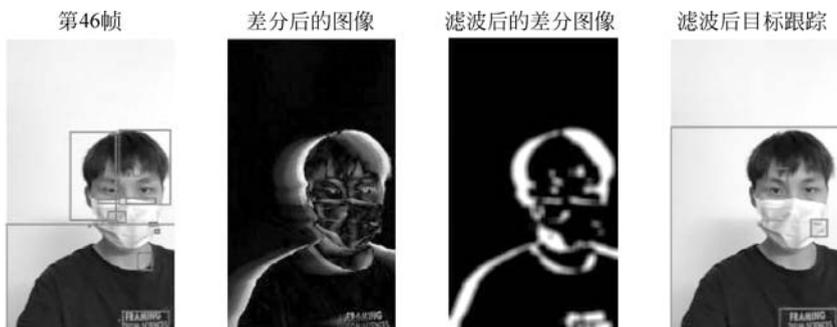


图 3.13 第 46 帧

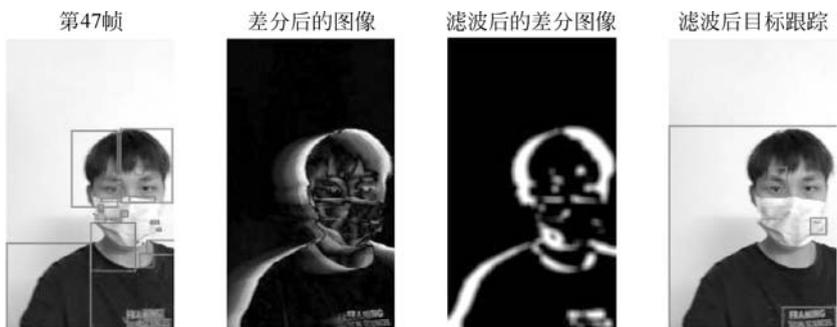


图 3.14 第 47 帧

3.4.3 实验结果分析

可以看到得出的实际结果和预想的已经基本吻合。未进行滤波的目标跟踪检测有很多噪点和小方框,滤波后的目标跟踪检测已经优化略去了许多噪点和微小的变动。完整的结果可以看保存的视频。本设计尝试将非局部均值滤波算法应用到了运动目标检测的研究中,这次尝试也成功对运动目标检测的结果进行了优化和改进,并且实际结果符合了预想的效果。

3.5 小结

通过对运动目标检测进行了对非局部均值滤波以及其快速算法的研究与应用,首先介绍了非局部均值滤波算法和运动目标检测的研究背景、意义以及常用的去噪算法,还列举了常用的运动目标检测方法。首先剖析了非局部均值滤波的原理以及该算法的不足之处,并且提出了基于积分图加速的快速非均值滤波算法以及运动目标检测之帧间差分法的分析。基于这些理论知识,将非均值滤波运用到运动目标检测中,对运动目标检测的图像序列进行

滤波去噪处理再进行运动目标检测,结果显示非均值滤波算法成功对运动目标检测的结果产生了比较好的改进及优化。总体来说,此次设计的实验结果说明非局部均值滤波去噪对运动目标检测的结果确实会有改良,在此基础上,可以进行更深更广的研究,希望非局部均值滤波算法还可以应用于更多的科学研究和实际应用。

参考文献

- [1] 梁广顺,汪日伟,温显斌.基于双边滤波与非局部均值的图像去噪研究[J].光电子:激光,2015(11): 2231-2235.
- [2] 肖本贤,陆诚,陈昊,等.基于帧间差分法和不变矩特征的运动目标检测与识别[C]//中国控制会议, 2008: 578-581.
- [3] WANG W F, DENG X Y, DING L, et al. Brain-inspired Intelligence and Visual Perception: The Brain and Machine Eyes[M]. Berlin: Springer, 2019.
- [4] 孙伟峰.基于非局部信息的信号与图像处理算法及其应用研究[D].济南:山东大学,2010.
- [5] 王斌. MATLAB实现数字图像增强处理[J].佳木斯大学学报,2005,23(1): 31-34.
- [6] 张丽果.快速非局部均值滤波图像去噪[J].信号处理,2013,029(008): 1043-1049.
- [7] 林洪文,涂丹,李国辉.基于统计背景模型的运动目标检测方法[J].计算机工程,2003(16): 101-103+112.
- [8] 魏宗坤.基于DM642的监控系统及运动目标检测算法的实现[D].成都:电子科技大学,2008.
- [9] 于成忠,朱骏,袁晓辉.基于背景差法的运动目标检测[C].全国自动化新技术学术交流会,2005.
- [10] 蔡斌.非局部均值去噪算法研究[D].合肥:中国科学技术大学,2015.
- [11] 刘悦.图像检测与识别的特征提取算法研究[D].哈尔滨:哈尔滨工程大学,2015.
- [12] 张爽,周慧鑫,牛肖雪,等.基于非局部均值滤波与时域高通滤波的非均匀性校正算法[J].光子学报,2013,43(1): 147-150.
- [13] 刘中合,王瑞雪,王锋德,等.数字图像处理技术现状与展望[J].计算机时代,2005(9): 6-8.
- [14] 范影乐,杨胜天,李轶. MATLAB仿真应用详解[M].北京:人民邮电出版社,2001.
- [15] 赵小川,赵斌. MATLAB数字图像处理:从仿真到C/C++代码的自动生成[M].北京:北京航空航天大学出版社,2015.
- [16] 吴福朝.计算机视觉中的数学方法[M].北京:科学出版社,2008.
- [17] 王秀芬,王汇源,王松.基于背景差分法和显著性图的海底目标检测方法[J].山东大学学报(工学版),2011,41(1): 12-16.
- [18] 林雯.新型基于帧间差分法的运动人脸检测算法研究[J].计算机仿真,2010(10): 248-251.
- [19] 张丹丹,娄焕.帧间差分法中阈值的选取[J].科技信息,2013(34): 204.
- [20] 陈虎,周朝辉,王守尊.基于数学形态学的图像去噪方法研究[J].工程图学学报,2004,25(2): 116-119.