

图像视频大作业一 Part III

2014011355 辛杭高

2017 年 5 月 4 日

本部分实验是针对视频的运动物体检测，我从视频中截取了 10 帧以图象格式存储，相邻的帧间隔 1 秒，存储于 src/img 文件夹中，作为运动物体检测的原材料。

1 Usage

本实验由 matlab 编码完成，在 exp3/src 下直接运行 exp3.m 即可，运行的时候必须保证 img 文件夹在同一目录，而且 result 文件夹也应该在这一目录 (result 可以是空文件夹，用于接收运动检测结果)。

输出结果为 9 个参数，前 4 个为使用 pixel matching 在保留 $1, \frac{1}{4}, \frac{1}{16}, \frac{1}{64}$ 比例的参数时进行匹配的平均 MSE，后 4 个为使用 DCT matching 在保留 $1, \frac{1}{4}, \frac{1}{16}, \frac{1}{64}$ 比例的参数时进行匹配的平均 MSE。如 64 dct match :296.4349 表示保留 $\frac{1}{64}$ 的参数进行匹配时各图的平均 MSE，最后一个参数为 Haar 小波匹配结果的 MSE。

最后还会 plot 两张图，表示两种方法下各帧 MSE 随帧数的变化。

2 Pixel Match

Pixel match 的原理为在给定位置的邻域内，对所有 $16 * 16$ 的像素块进行遍历，寻找在欧式距离意义下与原运动部分最接近的像素块。

表 1 中列出了各张图片标出运动物体的结果。

各帧 MSE 随帧数的变化图如图 1 所示。

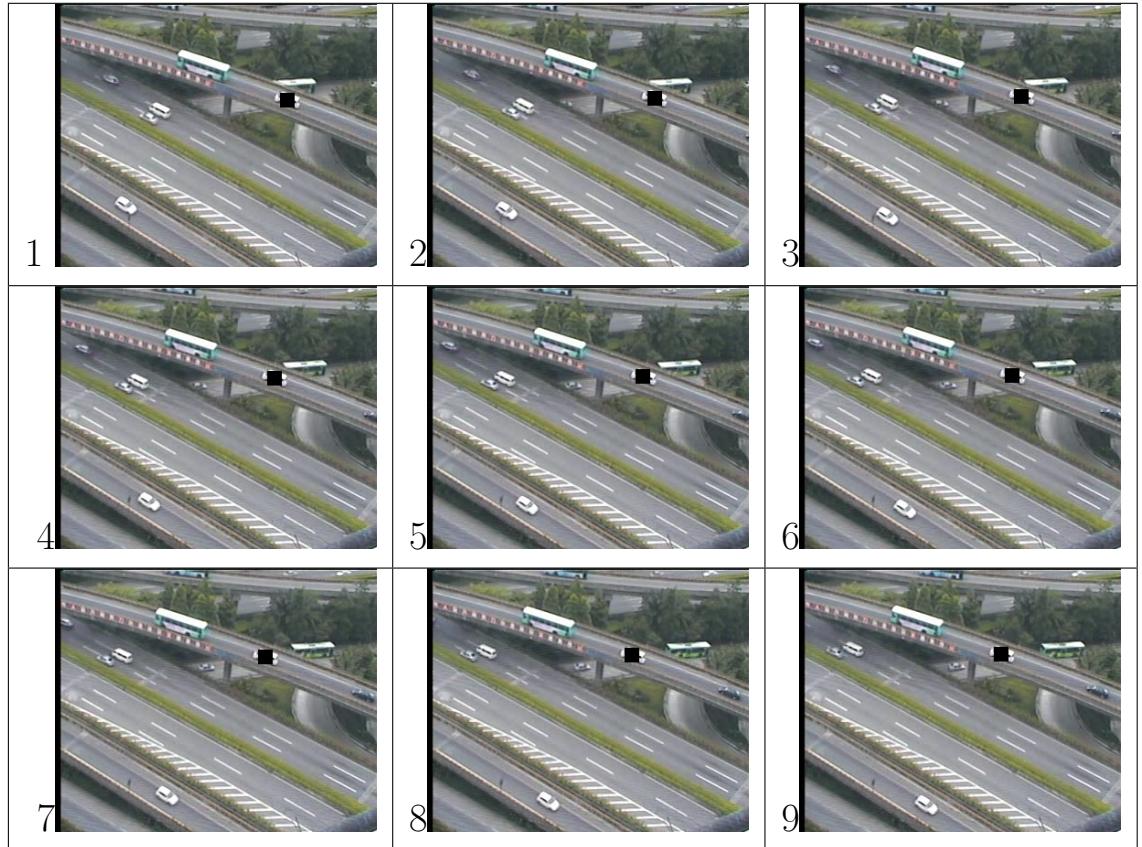


表 1: Pixel matching 在各帧上的结果 (用黑色标出).

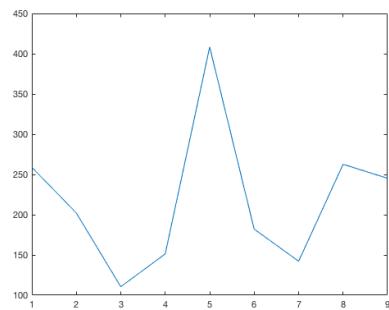


图 1: Pixel Match 下各帧 MSE 随帧数变化.

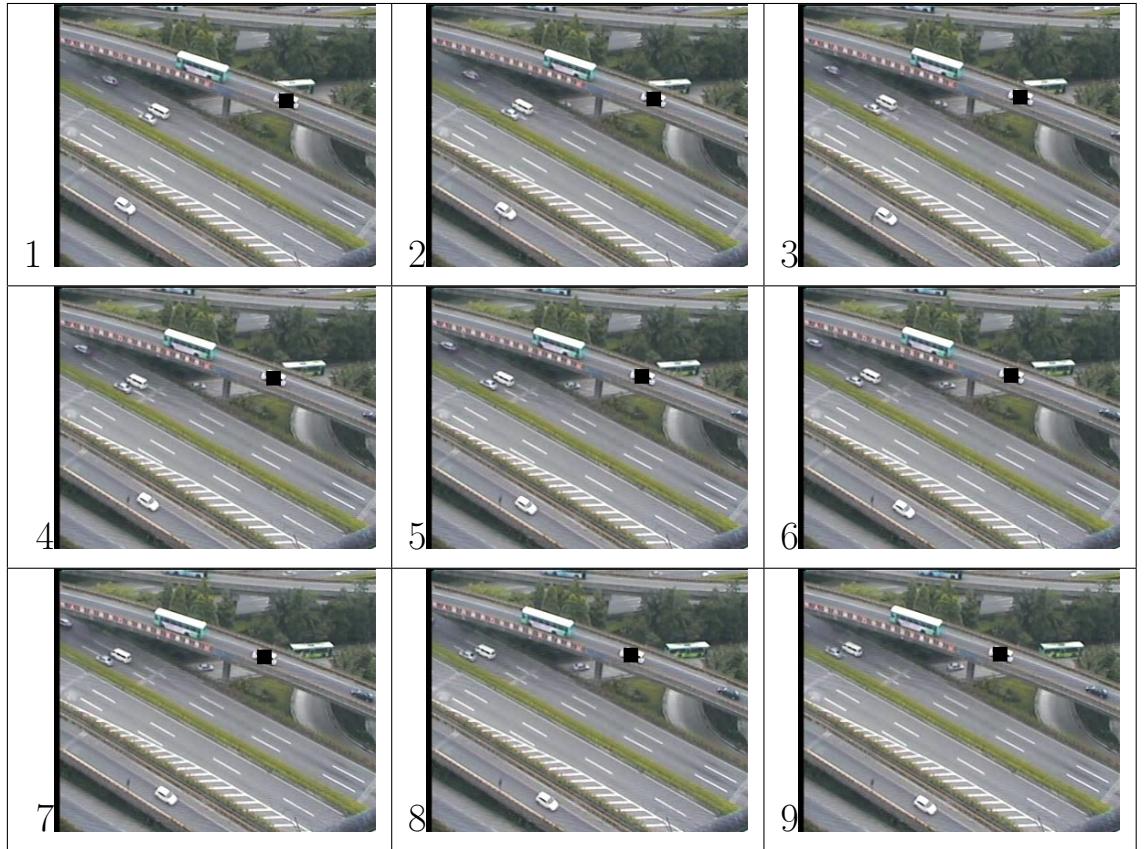


表 2: Dct matching 在各帧上的结果 (用黑色标出).

3 Dct Match

Compression domain block matching 的原理是在给定位置的邻域内，对所有 $16 * 16$ 的像素块进行遍历，对比每个像素块 DCT 之后的结果与原运动部分的距离，寻找距离最短的像素块。

表 2 中列出了各张图片标出运动物体的结果。

各帧 MSE 随帧数的变化图如图 2 所示。

4 Pixel Match 与 Dct Match 的对比

从图 1 和图 2 可以看出，两者在各个帧下的结果几乎完全一致，MSE 图也几乎一致。但理论上来讲，在保留所有参数的情况下 pixel match 的性

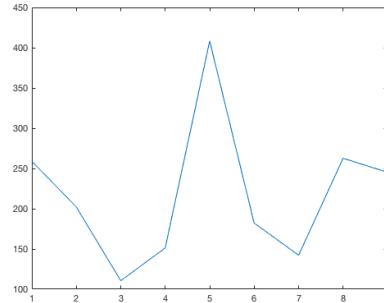


图 2: Dct Match 下各帧 MSE 随帧数变化.

能应该能在 MSE 的考量下达到最优。

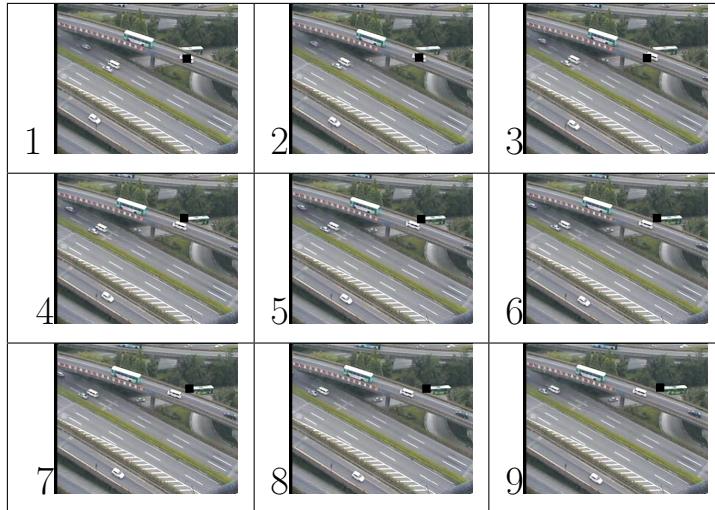
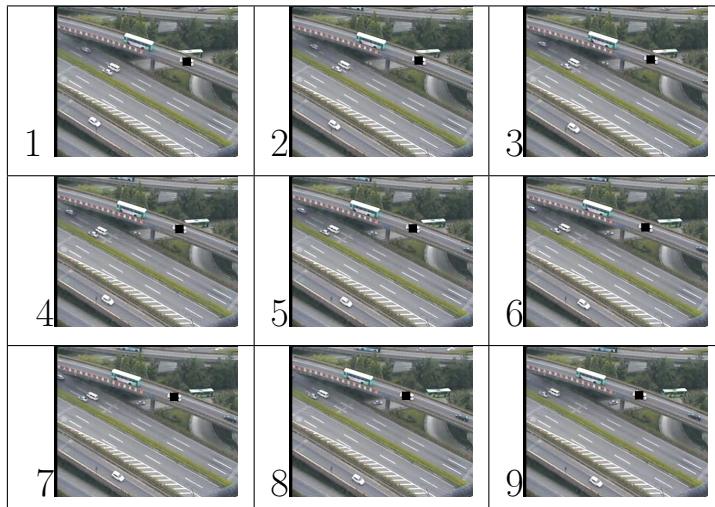
虽然像素序列可以和 DCT 序列相互转化，并且具有唯一性，但是由于数值精度的原因，使用像素序列匹配的结果应该在 MSE 指标下优于 DCT 后进行比较的结果。在本次实验中我选取了小车的中部，可以看出两个匹配都找出了小车的中部，但这也归功于 $16 * 16$ 的小块和小车的尺寸比较吻合，所以干扰比较小。

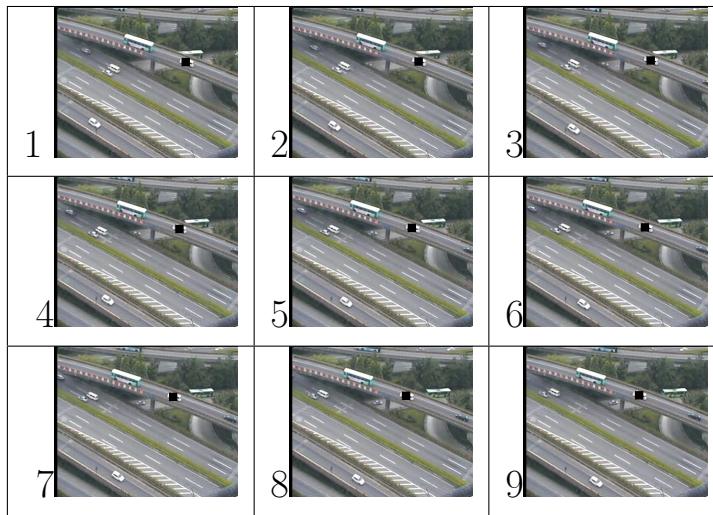
如果我们只选取部分参数进行匹配，对进行匹配的 9 个帧的 MSE 的平均值进行比较，可以获得

保留比例	Pixel Match	Dct Match
1	270.54	270.54
$\frac{1}{4}$	293.15	270.54
$\frac{1}{16}$	295.85	276.97
$\frac{1}{64}$	8777.36	302.14

可以看出，当丢弃一些数据进行匹配的时候 DCT 开始展示出明显的优势，在这种情况下 DCT 得到的结果将更符合运动物体运动轨迹，而 pixel match 已经开始出现明显错误。当然，这一结果也与我们的丢失策略有关系，在本实验中，我丢失 DCT 参数时集中丢弃了高频分量，所以对结果的影响不是很大。对于 Pixel 匹配，我采用的方式是隔若干行若干列保留一个像素点。

表 3 和表 4 列出了仅仅保留 $\frac{1}{64}$ 参数的情况下 Pixel 匹配和 DCT 匹配的差异。

表 3: 保留 $\frac{1}{64}$ 参数时 Pixel matching 结果.表 4: 保留 $\frac{1}{64}$ 参数时 Dct matching 的结果.

表 5: 保留 $\frac{1}{64}$ 参数时 Haar matching 的结果.

5 其他匹配算法—Haar 小波匹配算法

在 src/haarmatch.m 中我实现了 haar 小波匹配算法，与 DCT 不同的是，我们不再求 $16 * 16$ 小块的 DCT 变换，转而求每个小块的二维 haar 小波变换。在 matlab 中，haar 小波变换可以直接调用库函数

$$[a, h, v, d] = haart2(x) \quad (1)$$

针对 $16*16$ 的二维块，其结果中 a 为一个实数， h 包含一个 $8*8$ 的矩阵，一个 $4 * 4$ 的矩阵，一个 $2*2$ 的矩阵和一个实数， v 、 d 与 h 的规模相同。可以从其中选取某些参数作为我们的比较因子。

在本实验中，我选取了 4 个实数作为一个 $16*16$ 二维块的特征，来与之前 $\frac{1}{64}$ 比例的参数进行对比。对比后的结果显示在表 5 中。该结果的平均 MSE 为

$$MSE_{har} = 572.67 \quad (2)$$

从 MSE 的度量结果来看 haar 匹配的方法要略弱于 DCT 匹配的方法，但是从人眼视觉的角度来说，这两者几乎没有差别，另外在仅仅保留部分参数的情况下，haar 方法容易产生方形像素块，我们要检测的目标也正好是 $16 * 16$ 的像素块，在一定程度上影响了 haar match 的结果，如果检测的目标

变为其他形状应该会得到更好的结果。另外，haar 参数的选择也可以有其他的策略，可以匹配出更好的结果。