

反差式相机对焦系统原理归档

相机对焦

对焦就是通过驱动光学镜头或者焦平面来让物体在传感器上获得清晰地像。相机自动对焦分为主动式和被动式两种。主动式较早出现原理也很简单，主要是通过附加的设备确定对焦物体的距离，然后设定对焦即可。被动式较为复杂，主要分为反差式和相位式两种。相位式对焦主要是通过物体经过镜头边缘后在对焦面上成像，对焦面上的传感器根据像情况进行调整，速度快，但在低亮度下效果欠佳，同时为了提高效果需要较复杂的对焦点设计（双十字）。下面具体解释下反差对焦。

反差对焦原理

反差对焦，也叫对比度对焦，是使用图像中物体的反差进行对焦的算法，成像后根据成像效果决定下一步操作，这个类似人判断行为。反差对焦理论上比较科学，然而实际算法实现与图像反差计算算法，马达驱动算法，还有传感器刷新速度，驱动马达速度和精度，处理器的性能。反差对焦系统框图如图 1。

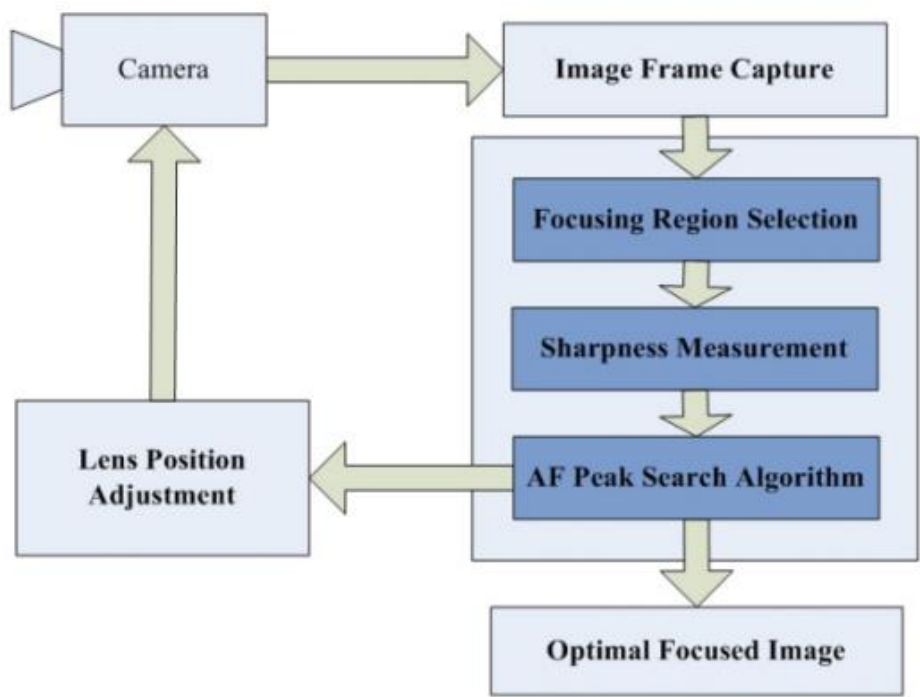


图 1 相机对焦系统框图

实际反差对焦时图像的发差值随着对焦的变化而变化，当对焦接近合焦的时候图像获得反差值是较大，反之当图像远离合焦时，反差值变小，这样我们就可以通过爬山算法进行对焦位置的搜索，具体效果可以见下图演示。当图像最清晰时，图像的反差值最高，当图像模糊远离正确对焦位置时，反差值较低，效果如图 2。

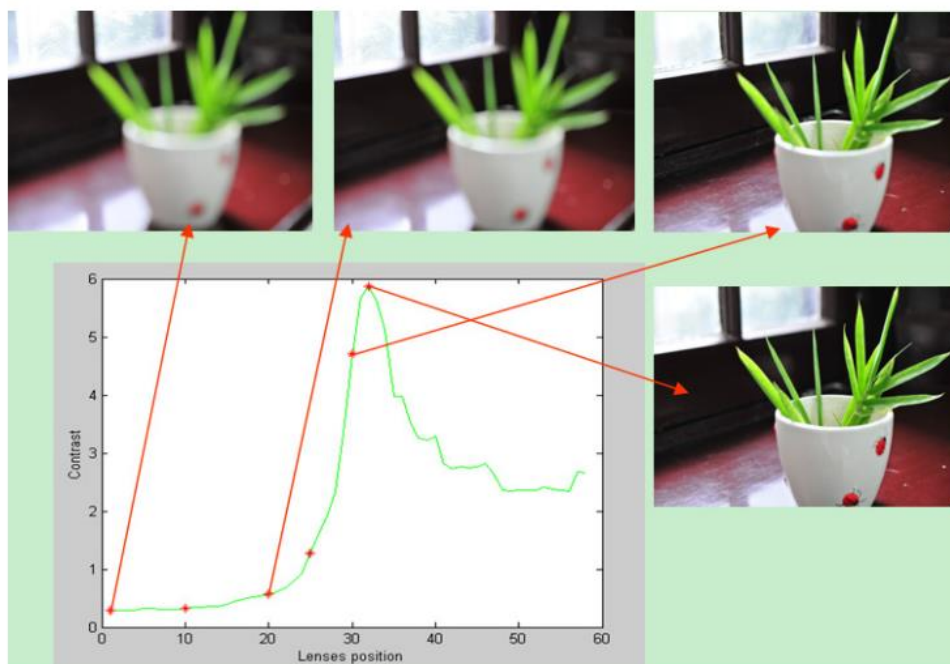
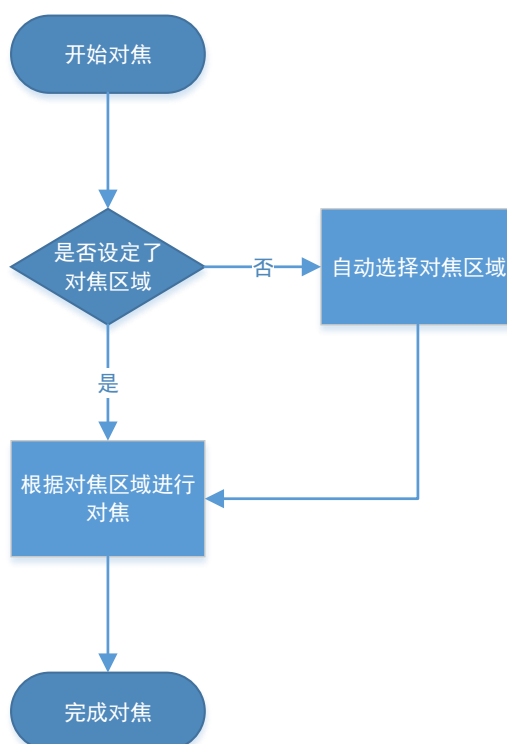


图 2 图像反差与对焦位置关系

算法详解

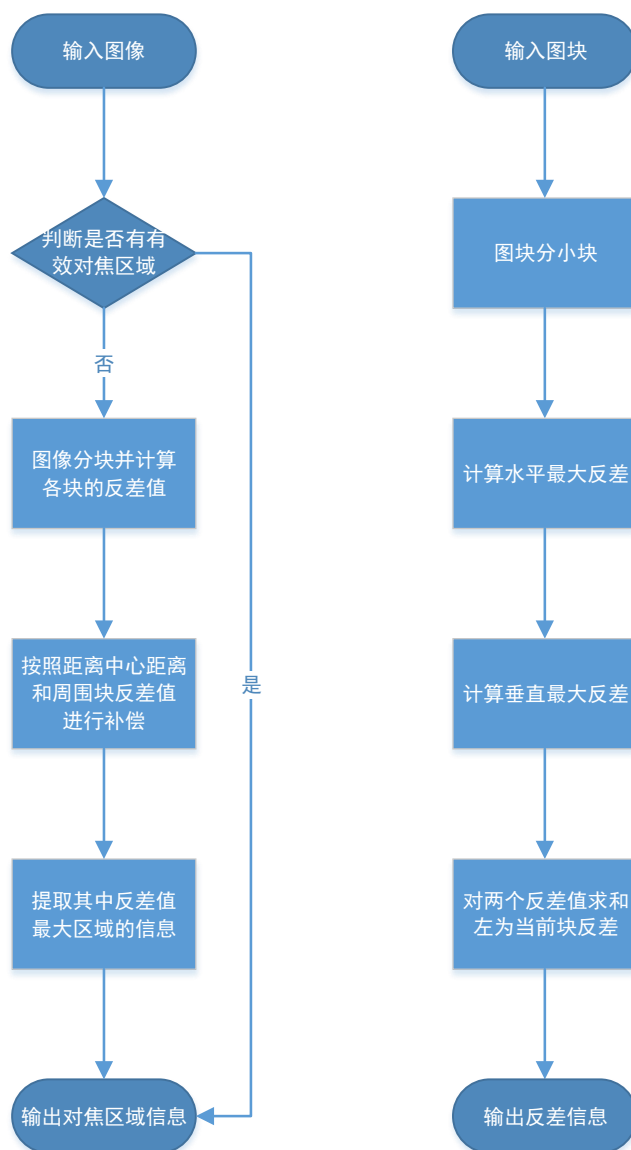
实际商业相机产品如果使用反差对焦，一般默认直接计算图像中心部分区域的对焦值，或者人为规定一个小的对焦窗，也可以通过根据图像本身的自适应的设定对焦区域。然后根据对焦区域进行对焦，合焦后结束对焦。总体流程图如流程图 1。



流程图 1 对焦算法整体流程

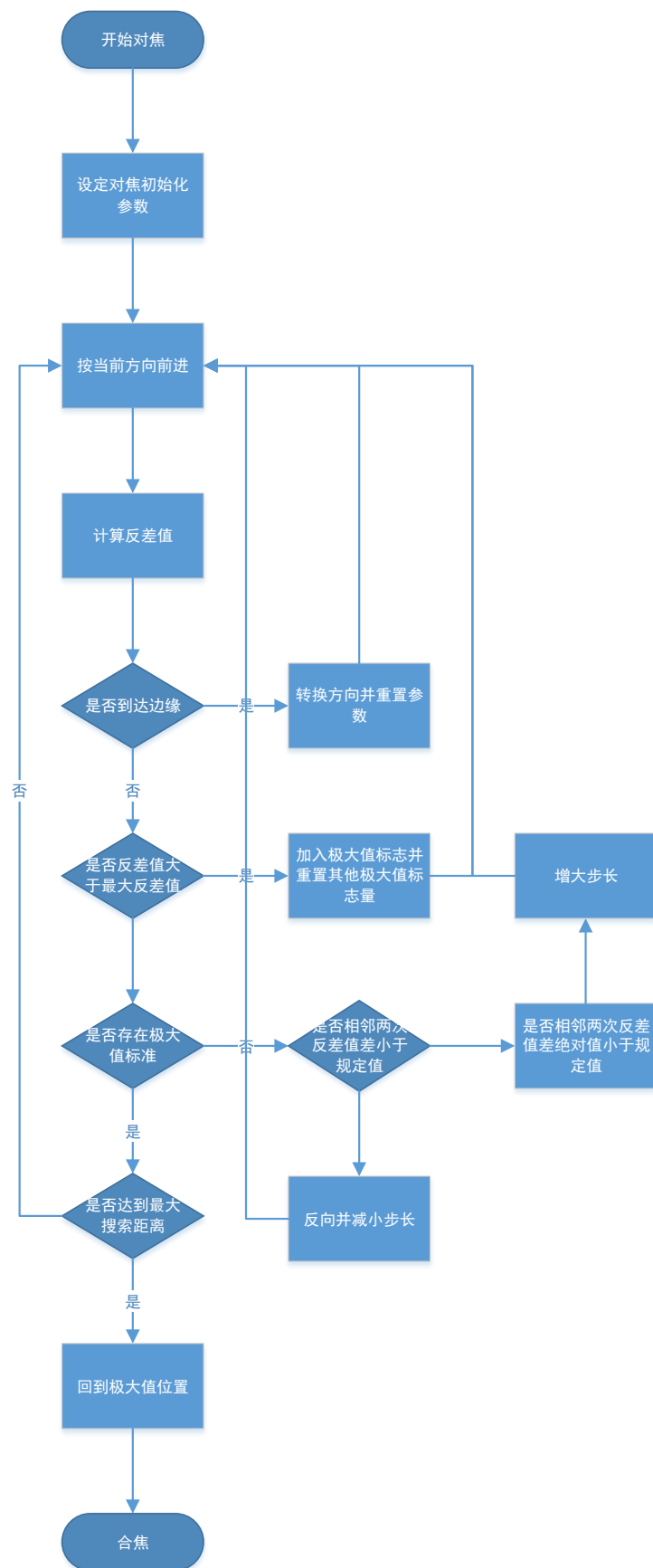
进入对焦流程，具体如流程图 2 左，关于对焦区域选择，可以自动选取，也可以人工设定，一般人工设定可以得到更好的效果，指定对焦区域我们可以 720p 环境下设定 128x128，1080p 可以设为更大，具体可以根据实际设定，注意判定对焦区域有效性主要看对焦区域不能太靠边缘，对焦区域不能太小。至于自动对焦框大小可以和手动设置的大小相仿。

按上面分块策略制定的大小，对图像进行分块计算反差值，比如 720p 图像分为宽 10，高 5 共 50 个块，分别计算反差值，注意计算反差值用的灰度图。计算各个分块反差值加入一定的中心偏重修正，具体为 dx ， dy 为分块中心到图像中心的距离，将 dx ， dy 分别除图像的半宽 640，半高 360 得到规则化的 ndx ， ndy ，中心权重为 $W_c = \frac{10000}{(80+10000 \times \sqrt{(ndx^2+ndy^2)})^2}$ 。



流程图 2 左：对焦区域选择 右：反差值计算

在计算完各个分块的反差值，得到各个分块的原始反差值 C ，为了得到更好的对焦效果，需要对反差值加权操作，如果分块周围有别的对焦块，则对四邻域的对焦块的反差值 $C/8$ 后加到当前对焦块上，如果没有四个相邻的对焦块，比如边缘的块只需要加相邻的两个或三个对焦块，得到新的反差值 C' 。最后对图像的反差值 C' 加权，得到距离加权后的图像反差值 $C'' = C' \times W_s \times W_c$ 。选取其中反差值最大区域作为对焦区域。得到对焦区域后进行爬山法搜寻最佳对焦位置。



流程图 3 爬山法对焦流程

至于计算反差值，流程如流程图 2 右，反差值计算为了简化运算量，使用块对比度，而不是常用的滤波后使用纹理特征。高斯滤波后使用 Canny 或者 Sobel 纹理对于富纹理的场景下效果较好，但是对于纹理较少噪声较多场景就不是很鲁棒了。将对焦图像块转为灰度图后分为 $n \times n$ 个小块，这里 128×128 分辨率图像块分为 8×8 个小块。即每个小块 16×16 ，分别计算每个 16×16 小块，左右两部分的亮度和之差和上下两部分的亮度和之差，分别取其中左右两部分的亮度和之差最大的值和上下两部分的亮度和之差最大的值作为该区域的反差值。

确定对焦区域后，按照爬山法进行对焦，爬山策略如流程图 3。首先重置所有参数。如果达到对焦两端则转向并恢复初始步长。如果新的反差值大于历史最大反差值则设置当前反差值为最大反差值，重置到反差值最大位置距离，和到反差值最大位置移动次数。如果存在历史最大值如果移动 3 次后没有更高的则回到历史最大值位置并合焦。如果本次反差减上次反差值小于负宽限值，则转向并减小步长。如果本次反差减上次反差值绝对值小于宽限值，则增加步长。算法中各个参数需要根据实际系统进行调整。

数据分析

图片集 1:



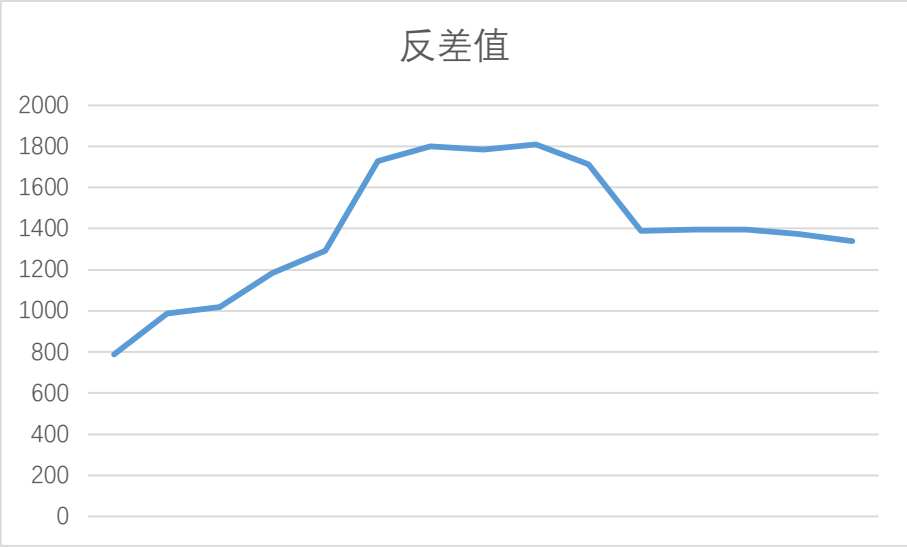
图 3 纹理丰富场景

如图 3，该图像数据集为窗外树叶等的数据，纹理极度丰富，拍摄时有轻微抖动。以下做数据分析。

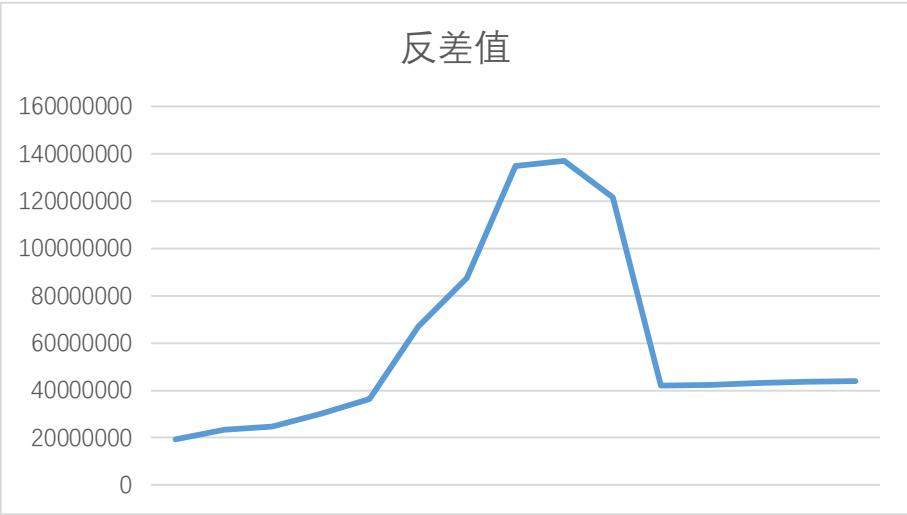


图 4 对焦区域选取与合焦距情况

自动对焦区域选择选取树木处区域，如图 4 左中的框为对焦位置，然后合焦后效果如图 4 右，可以看到对焦效果还是不错的。具体理论分析如图表 1，可以看到整个图像的反差值还是近乎凹函数走势的，由于这里采样稀疏，所以波动较大，但是使用爬山法进行搜索是没有问题的。左为对比，使用基于高斯滤波后使用边缘的方法处理，具体实现见 Nanda, H. & Cutler, R 的文章，边缘算法类似 Sobel 算子，稍微有所不同。如图表 2 是上述方法结果，可以看到在纹理丰富的条件下使用边缘信息处理效果也是很不错的，但是边缘信息计算量相对较高，时间复杂度高一些，而且没有质的提升。



图表 1 纹理丰富场景下对焦表现



图表 2 边缘滤波器方法效果

数据集 2:

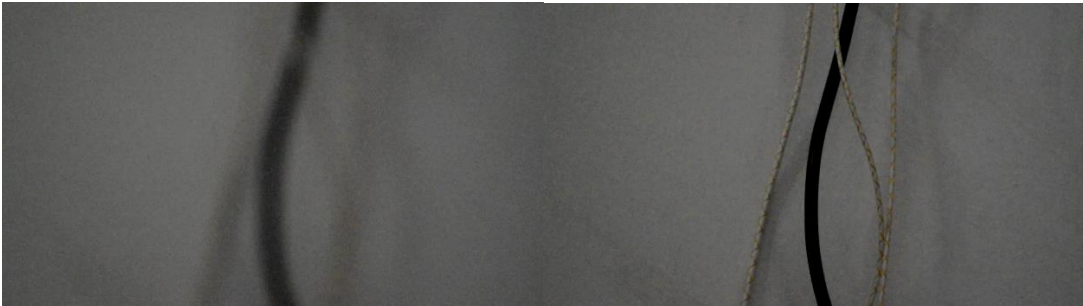


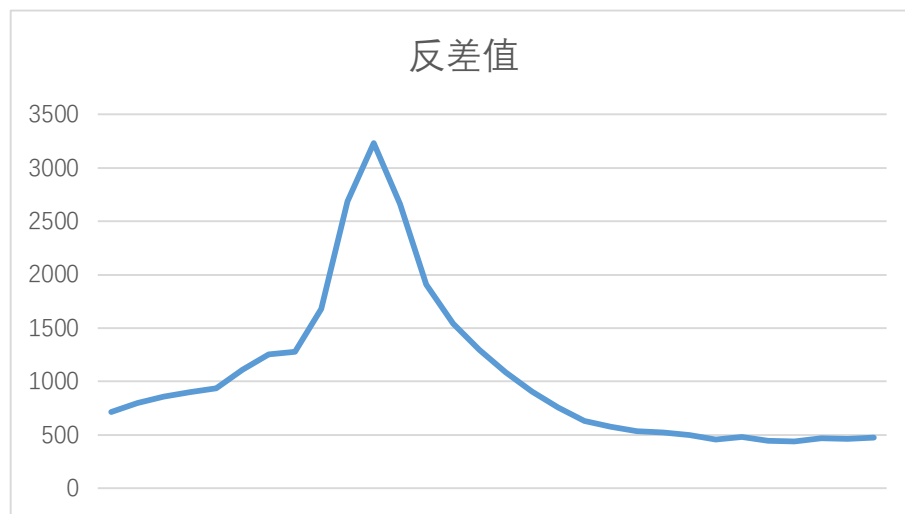
图 5 纹理不丰富且噪声较大场景

如图 5，由于亮度较低，该图有大量噪点，纹理部分较清楚，但是纹理区域不多，算是很坏的情况。这种情况相机也无法做到快速对焦，根据相机对焦行为，怀疑是全部搜索后提取最佳位置并控制线性驱动马达到达相应位置。按照本报告中方法提取对焦区域如图 6 左，合焦如图 6 右，可以看到选择了两条线之间反差较大的部分，合焦情况令人满意。

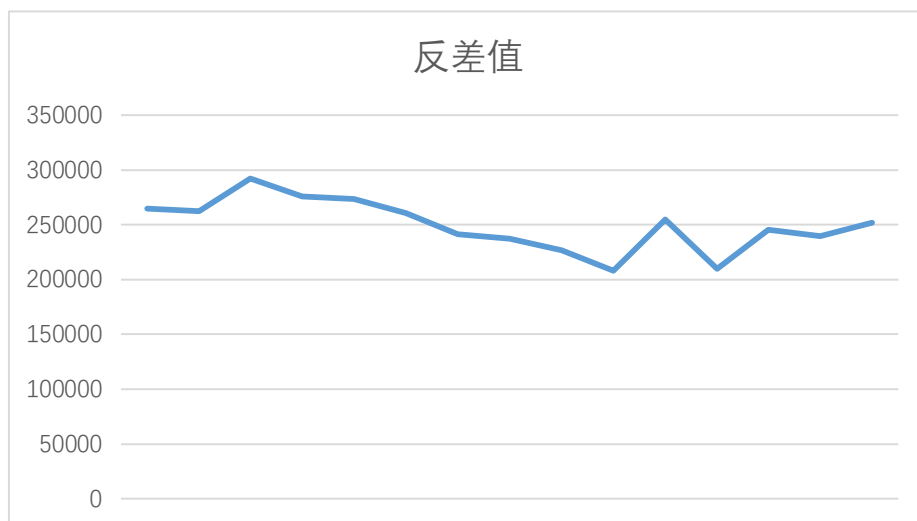


图 6 对焦区域选取与合焦距情况

具体分析数据，如图表 3、图表 4 可以看到高噪声少量纹理时，本报告算法较为鲁棒，选取了黑白交界的地方，而基于边缘的算法，由于噪声影响，结果不是很好，无法合焦。



图表 3 纹理缺少且噪声较大场景下对焦表现



图表 4 边缘滤波器方法效果

数据集 3:

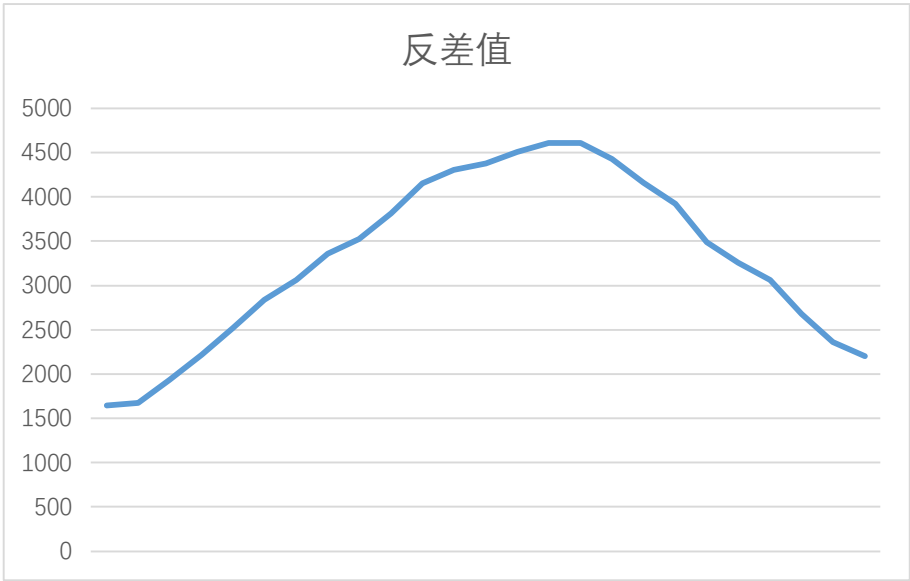
数据集 3 为一组有一定纹理并且有一定噪声的场景，该场景较接近实际应用，如图 7，图像有一定抖动，有大片平坦区域和边缘区域。如图 8 左为对焦区域选择情况，图 8 右为最终的合焦情况，可以看到对焦相对准确，整体清晰度不错。分析数据如图表 5 是反差值图，可以看到这种情况图像的反差值是一个较好的凹函数。实际在应用中可以很好的得到准确的对焦位置。



图 7 纹理较丰富有一定噪声的场景



图 8 对焦区域选择与合焦区域



图表 5 纹理正常且有一定噪声场景下对焦表现

总结

本报告提出了一种基于块反差的对焦值计算方法，并且表现出了较高的鲁棒性，特别是对于纹理不是很丰富，且噪声较严重的场景下也有不错的表现。相对于传统算法计算纹理，对于有噪声的场景需要先进行滤波，在高 ISO 条件下为了减少噪声，一些滤波算法可能导致边缘被抹掉，导致边缘不是很明显，使用联合双边滤波或导向滤波等虽然能保证边缘，但是大大增加了计算量，而频域的计算方式需要 FFT，计算代价也不低，本报告提出的方法，相对而言有较低的计算复杂度 $O(n)$ ，根据数据分析效果也可以接受，适合在嵌入式设备上运行。至于爬山算法的应用则依赖于设备具体参数，需要根据设备进行优化。