The Secant Method：正割法，又叫割线法、弦割法、弦法，是基于牛顿法的一种改进，基本思想是用弦的斜率近似代替目标函数的切线斜率，并用割线与横轴交点的横坐标作为方程式的根的近似。它是求解非线性方程的根的一种方法，属于逐点线性化的方法。

Cardboard反畸变代码中用到的正割法：

public float distortInv(float radius) {

// Secant method.

float r0 = 0;

float r1 = 1;

float dr0 = radius - distort(r0);

while (Mathf.Abs(r1 - r0) > 0.0001f) {

float dr1 = radius - distort(r1);

float r2 = r1 - dr1 \* ((r1 - r0) / (dr1 - dr0));

r0 = r1;

r1 = r2;

dr0 = dr1;

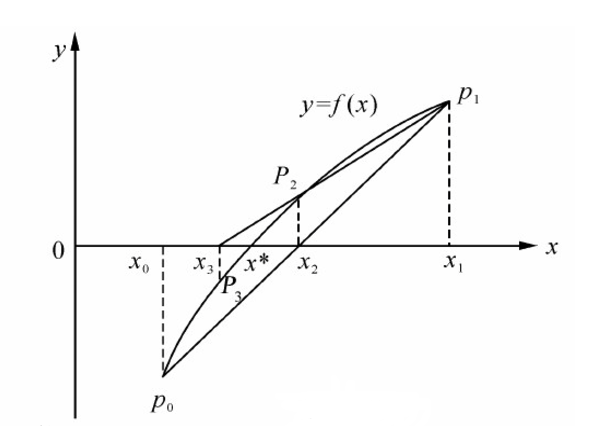
}

return r1;

}

正割法迭代公式：

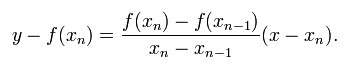




在数值分析中，割线法是一个求根算法，该方法用一系列割线的根来近似代替函数f的根。

方法的推导

给定*xn*−1和*xn*，我们作通过点(*xn*−1, *f*(*xn*−1))和(*xn*, *f*(*xn*))的直线，如右图所示。注意这条直线是函数*f*的割线，或弦。这条割线的点斜式直线方程为：



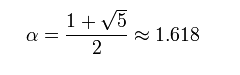
我们现在选择xn+1为这条割线的根，因此xn+1满足以下的方程：



解这个方程，便可以得出割线法的递推关系。

**收敛**

如果初始值*x*0和*x*1离根足够近，则割线法的第*n*次迭代*x*收敛于*f*的一个根。收敛速率为α，其中：



是黄金比。特别地，收敛速率是超线性的。

这个结果只在某些条件下才成立，例如f是连续的二阶可导函数，且函数的根不是重根。

如果初始值离根太远，则不能保证割线法收敛。