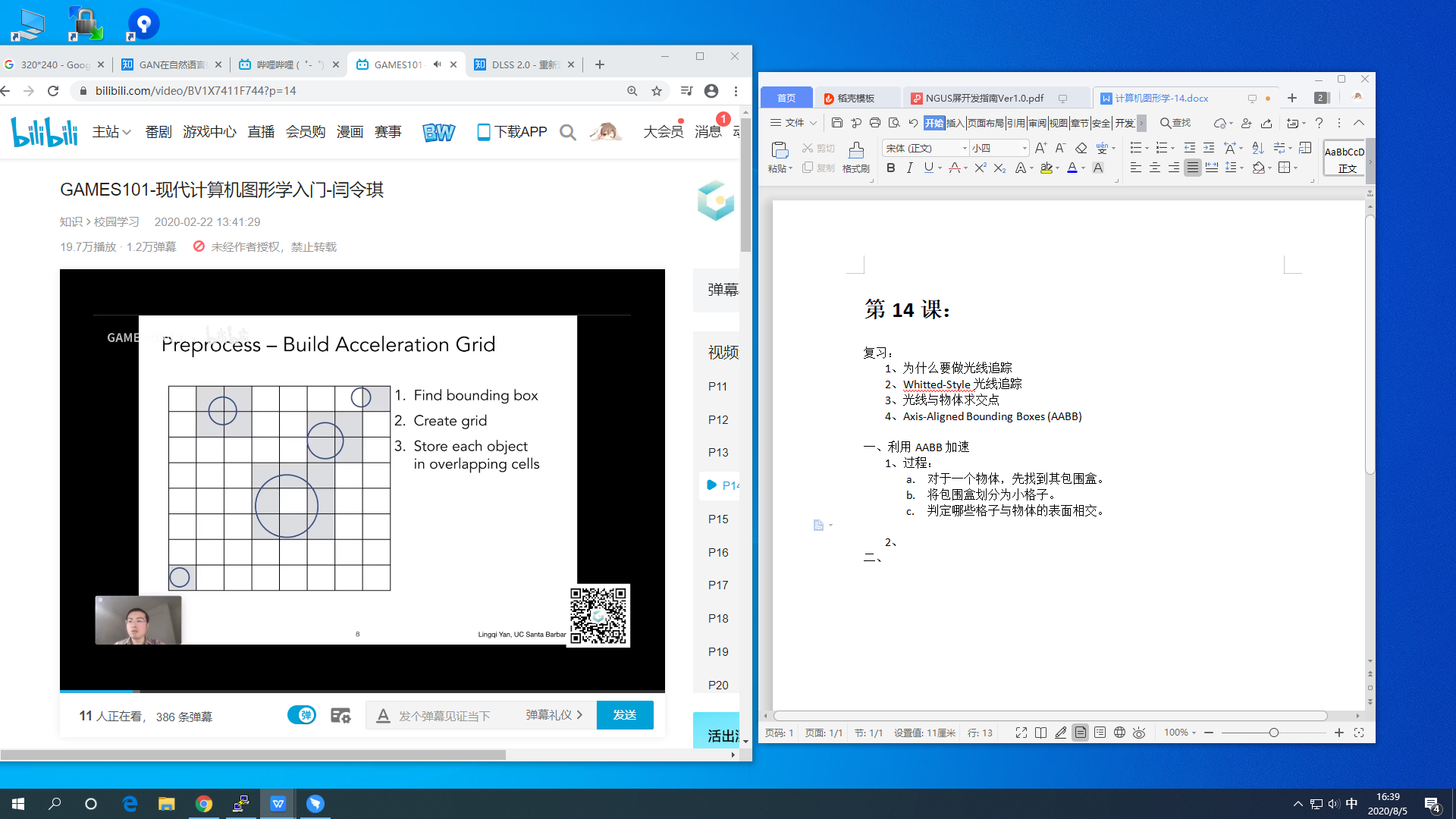
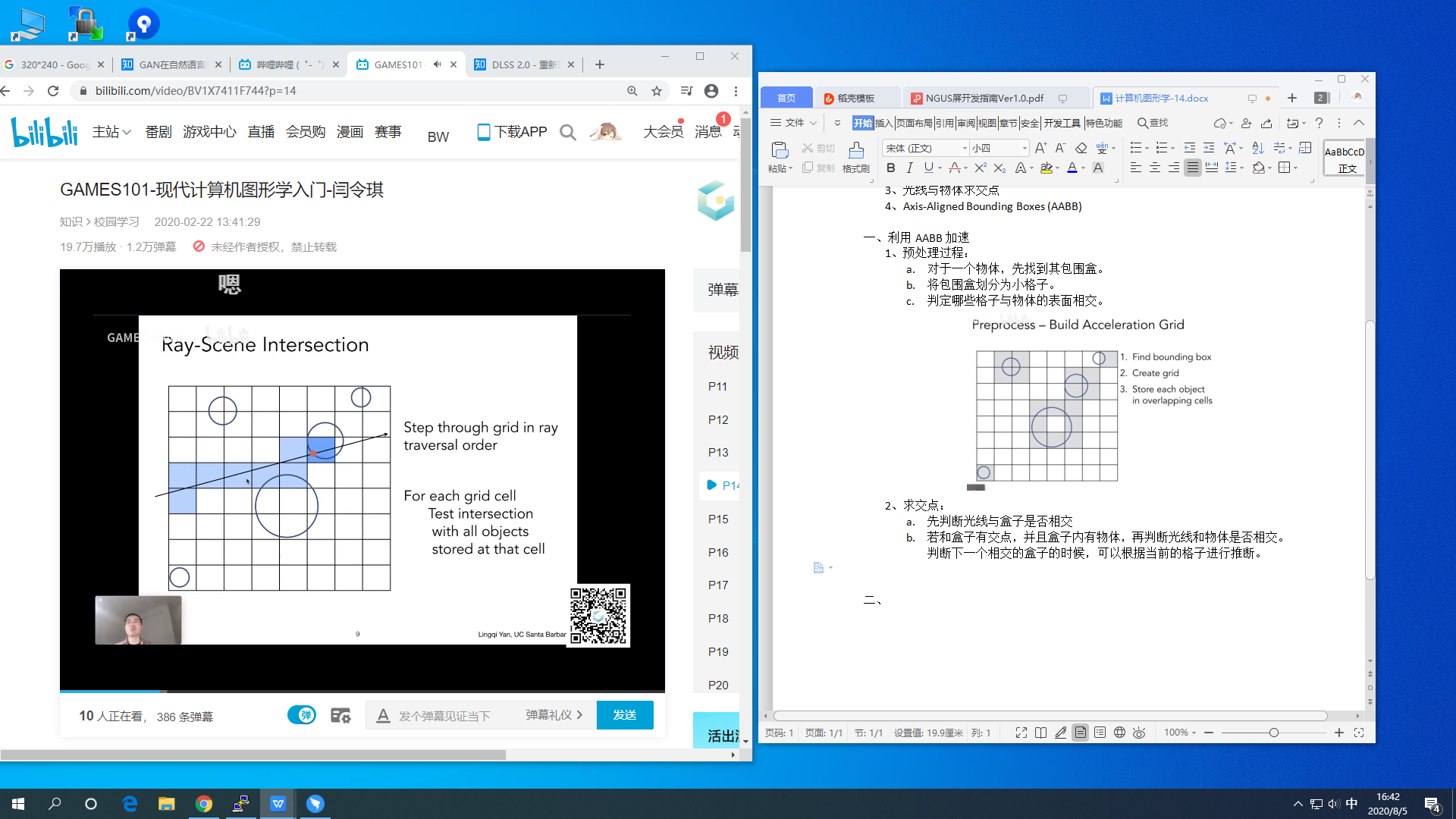
# 第14课：

复习：

1. 为什么要做光线追踪
2. Whitted-Style光线追踪
3. 光线与物体求交点
4. Axis-Aligned Bounding Boxes (AABB)
5. 利用AABB加速
6. 预处理过程：
7. 对于一个物体，先找到其包围盒。
8. 将包围盒划分为小格子。
9. 判定哪些格子与物体的表面相交。

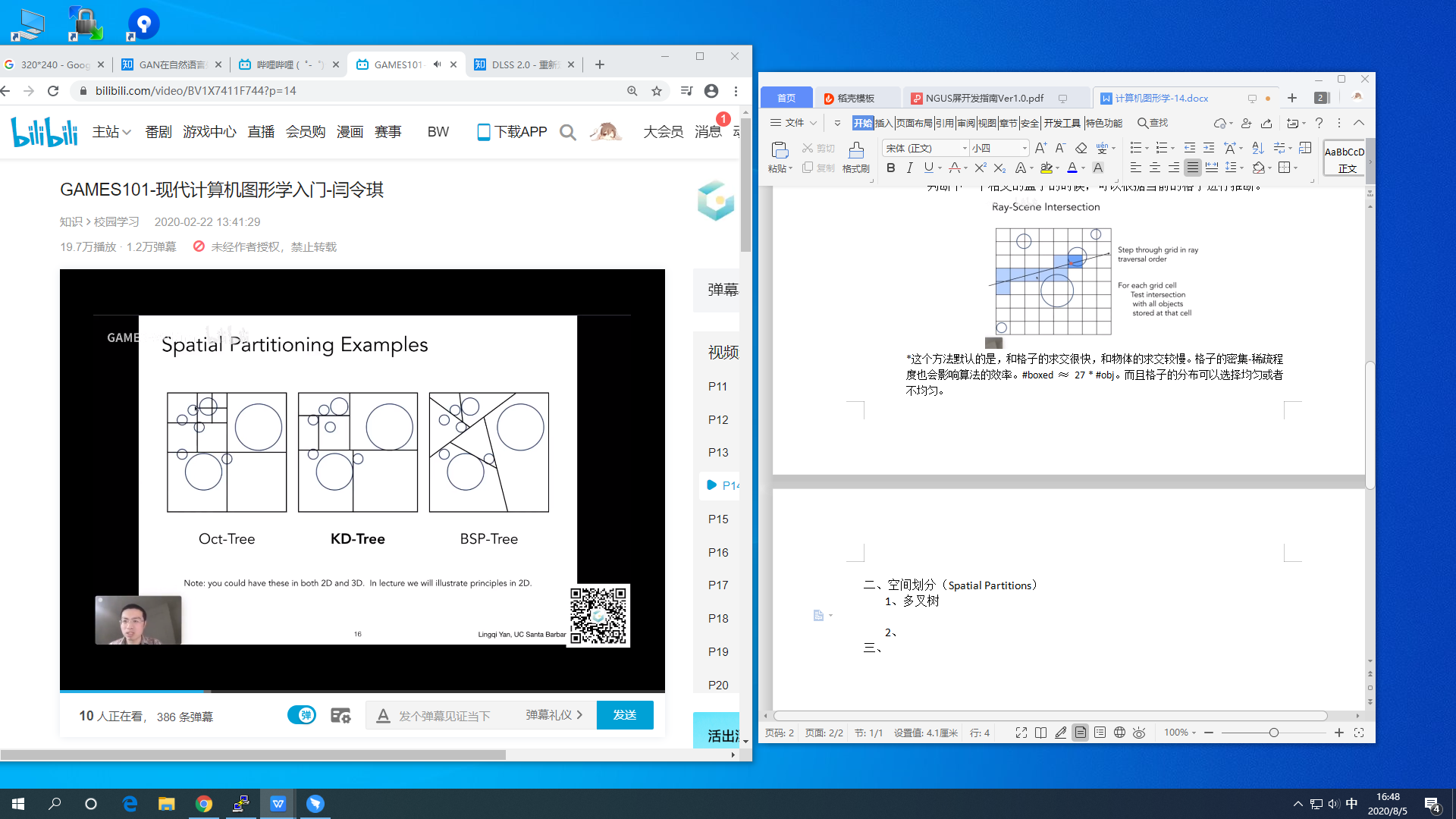


1. 求交点：
2. 先判断光线与盒子是否相交
3. 若和盒子有交点，并且盒子内有物体，再判断光线和物体是否相交。判断下一个相交的盒子的时候，可以根据当前的格子进行推断。

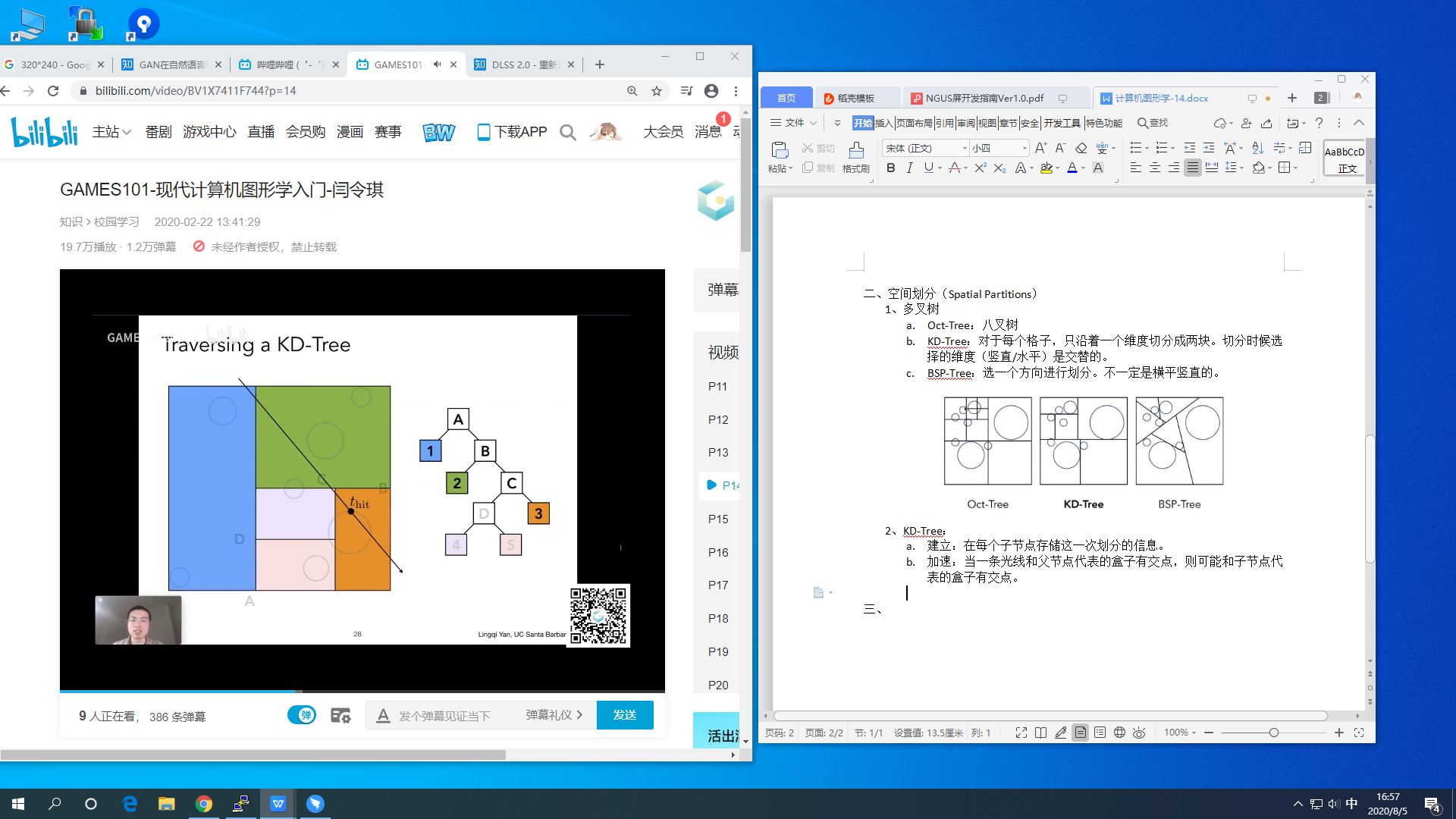


\*这个方法默认的是，和格子的求交很快，和物体的求交较慢。格子的密集-稀疏程度也会影响算法的效率。#boxed ≈ 27 \* #obj。而且格子的分布可以选择均匀或者不均匀。

1. 空间划分（Spatial Partitions）
2. 多叉树
3. Oct-Tree：八叉树
4. KD-Tree：对于每个格子，只沿着一个维度切分成两块。切分时候选择的维度（竖直/水平）是交替的。
5. BSP-Tree：选一个方向进行划分。不一定是横平竖直的。



1. KD-Tree：
2. 建立：在每个子节点存储这一次划分的信息。
3. 加速：当一条光线和父节点代表的盒子有交点，则可能和子节点代表的盒子有交点。

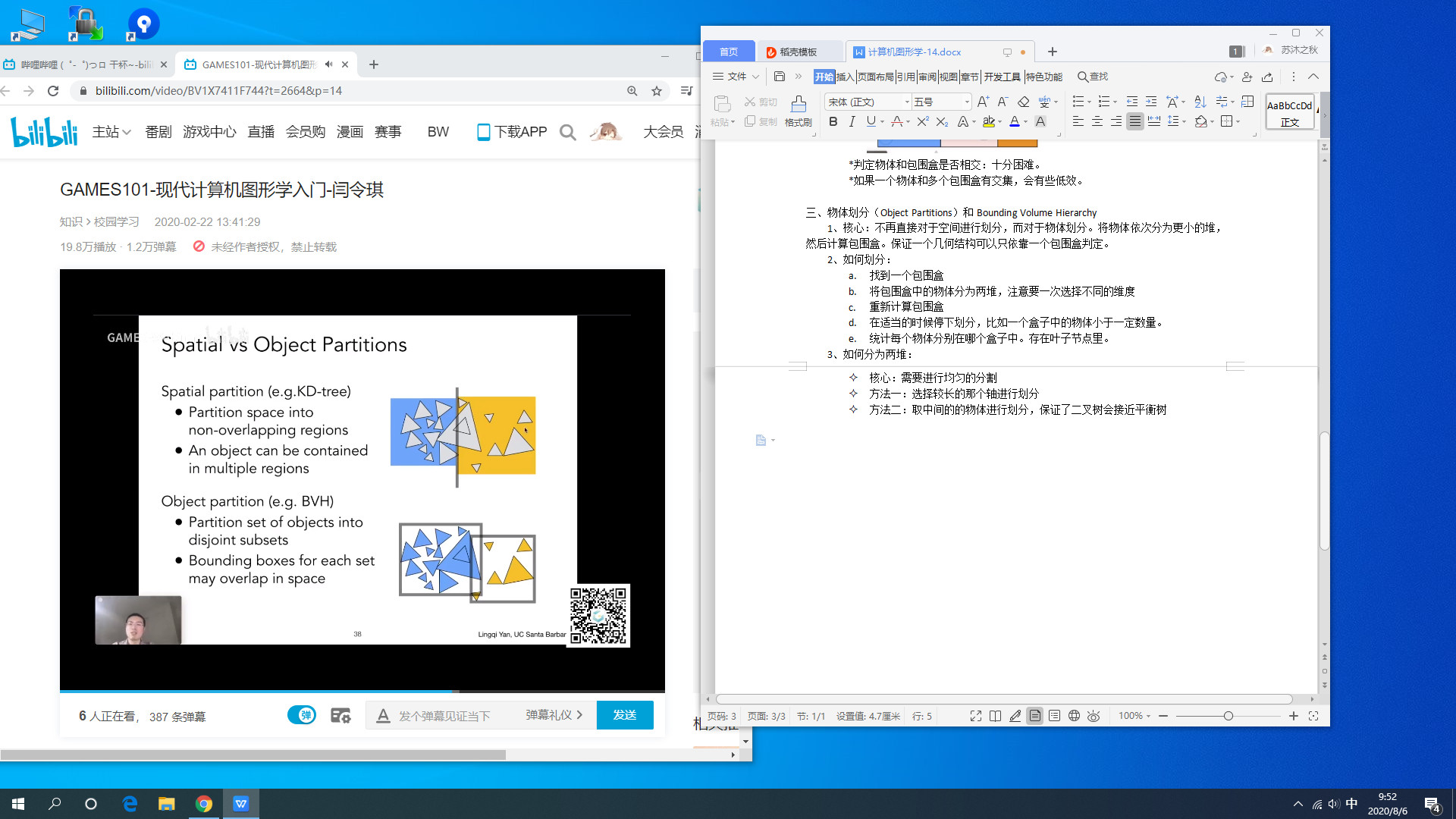


\*判定物体和包围盒是否相交：十分困难。

\*如果一个物体和多个包围盒有交集，会有些低效。

1. 物体划分（Object Partitions）和Bounding Volume Hierarchy
2. 核心：不再直接对于空间进行划分，而对于物体划分。将物体依次分为更小的堆，然后计算包围盒。保证一个几何结构可以只依靠一个包围盒判定。
3. 如何划分：
4. 找到一个包围盒
5. 将包围盒中的物体分为两堆，注意要一次选择不同的维度
6. 重新计算包围盒
7. 在适当的时候停下划分，比如一个盒子中的物体小于一定数量。
8. 统计每个物体分别在哪个盒子中。存在叶子节点里。
9. 如何分为两堆：

* 核心：需要进行均匀的分割
* 方法一：选择较长的那个轴进行划分
* 方法二：取中间的的物体进行划分，保证了二叉树会接近平衡树



1. 辐射度量学
2. 引入原因：

* 在Blilnn-Phong模型中，有光强I，这个I的物理意义是什么？
* Whitted-Style风格光线追踪存在一定的问题

1. 定义：

Radiant flux, intensity, irradiance, radiance

Radiant Energy：电磁辐射的能量Q，单位是焦耳[J]

Radiant flux：单位时间内的能量，单位是瓦特[Watt]， dQ / dt

Radiant Intensity: power per unit solid angle(立体角)。单位是坎德拉[candelas]

\* 立体角类似于弧度制在球体上的衍生。立体角是球面上的单位面积除以半径平方。 dw = S / r2

