CS 112 - Collision Detection
Objects have millions of triangles

For two objects with \( m \) and \( n \) triangles
- You need \( mn \) triangle-triangle intersections
- \( 10^{12} \) intersection computations for just \( \text{two} \) objects

Dynamic scenes – 30 frames per second
- Humungous computation needs

Need to make it efficient
Efficiency Measures

- Most of the time objects do not intersect
  - Fast rejections
  - Spend time on intersection computations only when objects intersect
- Two important issues
  - Bounding Volume – How closely it approximates the object?
  - Intersection Computation – How simple are the intersection computations?
Bounding Volumes

- Enclose the object
- The ratio of the object volume to the bounding volume should be as close to 1 as possible
- Depends on the shape of the object
Bounding Volumes

- **Axis-aligned**
  - The planes of the box is aligned with the *world coordinates*

- **Object oriented**
  - The planes are aligned to hug the object more closely
  - More rejections
Bounding Volumes

- Spherical
  - Enclosing *sphere*

- Spherical Shells
  - Between *concentric spherical* shells

- Convex Hull
  - Closest Fit (Optimal)
  - Smallest Ratio
Intersection Calculations

- **Axis aligned Bounding Box**
  - Compare min and max in X, Y and Z directions
  - If all of them intersect, then the object intersects

- **Spherical**
  - Find the distance between the spheres
  - If less than the summation of the radius, then intersects
Intersection Computations

- Object Oriented and Spherical Shells
  - Complex computations
  - Google for reference

- Convex Hull
  - Convex hull of the object is the object itself
  - Therefore, need exhaustive triangle-triangle computation
Updating the bounding boxes

- Axis aligned Bounding box
  + Translation invariant
  - Any other kind of movements, box no longer remains axis-aligned
  - Needs to be recomputed frame by frame
  + Very simple computation
Updating the bounding boxes

- **Spherical**
  - Transformation invariant
  - Simple intersection computation
  - Lot of empty space in the volume

- **Oriented Bounding Box**
  - Transformation invariant
  - Complex intersection computation
  - Compact volume
Hierarchical Bounding Volumes

- Similar to spatial subdivision
- But for each object
- Slightly different
  - Union of children may not encompass the parent
Hierarchical Bounding Volumes

- If does not intersect, do not explore the children.
- If intersects, do bounding volume intersection on children.
- Continue till you get to the triangle-triangle intersection.
  - Very few of them needs to be computed.