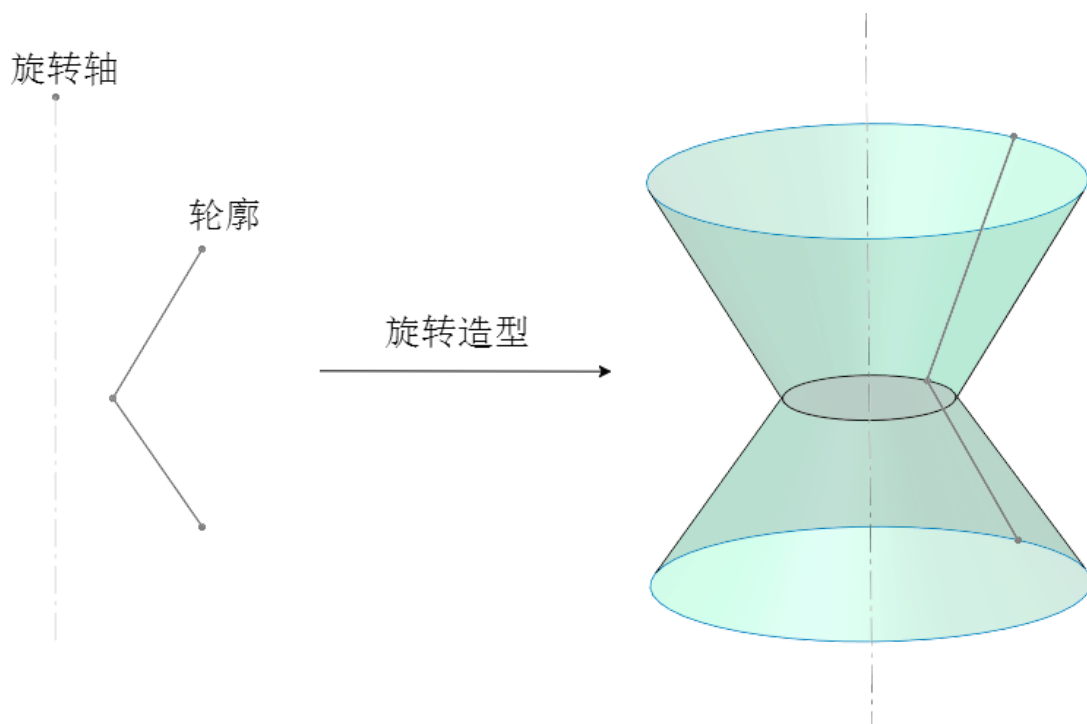


🟡 10. 旋转成型LatheGeometry

生活中有很多的几何体具备旋转特征，three.js提供了一个类 `LatheGeometry()`，`LatheGeometry` 可以利用一个2D轮廓，经过旋转变换生成一个3D的几何体曲面。



旋转 `LatheGeometry` 案例

格式: `LatheGeometry(points, segments, phiStart, phiLength)`

js

参数	值
points	Vector2表示的坐标数据组成的数组
segments	圆周方向细分数，默认12
phiStart	开始角度,默认0
phiLength	旋转角度，默认 2π

- `LatheGeometry` 类第一个参数就是旋转轮廓，旋转轮廓使用多个二维向量 `Vector2` 表示的xy坐标去描述。

- `LatheGeometry` 的旋转轮廓默认绕y轴旋转生成曲面几何体。

```
// Vector2表示的三个点坐标，三个点构成的轮廓相当于两端直线相连接
const pointsArr = [
  new THREE.Vector2(50, 60),
  new THREE.Vector2(25, 0),
  new THREE.Vector2(50, -60)
];
// LatheGeometry: pointsArr轮廓绕y轴旋转生成几何体曲面
// pointsArr: 旋转几何体的旋转轮廓形状
const geometry = new THREE.LatheGeometry(pointsArr);
```

js

旋转细分精度

```
// 30: 旋转圆周方向几何体细分精度
const geometry = new THREE.LatheGeometry(pointsArr, 30);
```

js

旋转角度

```
// 0, Math.PI: 旋转的开始角度和结束角度
const geometry = new THREE.LatheGeometry(pointsArr, 30, 0, Math.PI);
```

js

曲线生成旋转轮廓

通过二维样条曲线 `SplineCurve` 生成一个光滑的曲线旋转轮廓。

```
// 通过三个点定义一个二维样条曲线
const curve = new THREE.SplineCurve([
  new THREE.Vector2(50, 60),
  new THREE.Vector2(25, 0),
  new THREE.Vector2(50, -60)
]);
// 曲线上获取点, 作为旋转几何体的旋转轮廓
const pointsArr = curve.getPoints(50);
console.log('旋转轮廓数据', pointsArr);
// LatheGeometry: pointsArr轮廓绕y轴旋转生成几何体曲面
const geometry = new THREE.LatheGeometry(pointsArr, 30);
```

js

