EAI Cheatsheet by 卓致用，杨涛瑜，张煦恒

1 Robotics

FK：正向运动学给定关节角度，计算末端执行器的位置和姿态。IK：给定正向运动学 $T_{s\to e}(\theta)$ 和目标姿态 $T_{target}=\mathbb{SE}(3)$，求解满足以下条件的关节角度 θ：$T_{s\to e}(\theta)=T_{target}$。IK 比 FK 更复杂，因为 T^{-1} 可能很难计算，所以 **通常可能多解或无解**。

三维空间中 (R,t)“完全自由度”配置。至少 6 个自由度可以保证覆盖此空间，从而 IK 有解（但有时候可能得不到解析解，只能得到数值解）。*引理：如果机械臂构型满足 Pieper Criterion，则有解析解（闭式解）*。6DoF 保证有解，但是这个解可能超出了可行空间（如碰撞解），所以额外增加 1 冗余形成 7 DoF，可扩大解空间，更有可能找到可行解。但一味增加自由度会带来工程复杂性并延长反应时间。目前工业界一般 6 或者 7 DoF。

欧拉角：$R_x(\alpha) := \begin{bmatrix}1&0&0\\0&\cos\alpha&-\sin\alpha\\0&\sin\alpha&\cos\alpha\end{bmatrix}, R_y(\beta) := \begin{bmatrix}\cos\beta&0&\sin\beta\\0&1&0\\-\sin\beta&0&\cos\beta\end{bmatrix}, R_z(\gamma) := \begin{bmatrix}\cos\gamma&-\sin\gamma&0\\\sin\gamma&\cos\gamma&0\\0&0&1\end{bmatrix}$

任意旋转均可拆为 $R=R_z(\alpha)R_y(\beta)R_x(\gamma)$。这个顺序可以变，但一般默认是这个顺序。问题：1. 对于一个旋转矩阵，**其欧拉角可能不唯一**。2. **Gimbal Lock**：如果三次旋转中第二次旋转 β 的角度为 $\pi/2$，那么剩下 2 个自由度会变成 1 个。

欧拉定理/axis-angle：任意三维空间中的旋转都可以表示为绕一个固定轴 $\hat{\omega}\in\mathbb{R}^3$（单位向量，满足 $\|\hat{\omega}\|=1$）旋转一个正角度 θ 的结果。问题：**不唯一性**：$(\hat{\omega},\theta)$ 和 $(-\hat{\omega},-\theta)$ 代表同一个旋转；$\theta=0$ 时对应 $R=I$，任意 $\hat{\omega}$ 都行；$\theta=\pi$ 时，绕轴 $\hat{\omega}$ 和绕轴 $-\hat{\omega}$ 旋转 π 得到的结果是相同的。这种情况对应 $\mathrm{tr}(R)=-1$。将旋转角 θ 限制在 $(0,\pi)$ 内，那么对于大部分旋转，其轴角表示就是唯一的（不考虑不旋转、旋转 π）。

对于一个单位轴向量（axis）$\mathbf{u}=[x,y,z]^\top$，其对应的叉乘矩阵（cross product matrix）$K=\begin{bmatrix}0&-z&y\\z&0&-x\\-y&x&0\end{bmatrix}$。

Rodrigues 旋转公式（向量形式）：向量 $\mathbf{v}$ 沿着单位向量 $\mathbf{u}$ 旋转 θ 角度之后的向量 $\mathbf{v}'$ 为 $\mathbf{v}'=\cos(\theta)\mathbf{v}+(1-\cos(\theta))(\mathbf{u}\cdot\mathbf{v})\mathbf{u}+\sin(\theta)(\mathbf{u}\times\mathbf{v})$（**矩阵形式**）绕单位轴 $\mathbf{u}$ 旋转 θ 的旋转矩阵 R_θ 可以表示为 $R_\theta=I+(1-\cos\theta)K^2+\sin\theta\cdot K=e^{\theta K}$。*证明*：拆开级数然后用 $K^2=\mathbf{u}\mathbf{u}^\top-I, K^3=-K$

从旋转矩阵 R 反求 $(\hat{\omega},\theta)$：当 $\theta\in(0,\pi)$ 时，$\theta=\arccos\frac{1}{2}[\mathrm{tr}(R)-1]$，$[\hat{\omega}]=\frac{1}{2\sin\theta}(R-R^\top)$。定义两个旋转矩阵之间的 **旋转距离**：从姿态 R_1 转到姿态 R_2 所需的最小旋转角度。两个旋转的关系是：$(R_2R_1^\top)R_1=R_2$。旋转距离：$\mathrm{dist}(R_1,R_2)=\theta(R_2R_1^\top)=\arccos\left(\frac{1}{2}\left[\mathrm{tr}(R_2R_1^\top)-1\right]\right)$

Quaternion：$q=w+xi+yj+zk$。其中 w 实部，x,y,z 虚部。i,j,k 是虚数单位，满足 $i^2=j^2=k^2=ijk=-1$ 和反交换性质（**没有交换律**）：$ij=k=-ji,\quad jk=i=-kj,\quad ki=j=-ik$。可以表示为向量形式 $q=(w,\mathbf{v}),\quad \mathbf{v}=(x,y,z)$。

乘法：$q_1q_2=(w_1w_2-\mathbf{v}_1^\top\mathbf{v}_2, w_1\mathbf{v}_2+w_2\mathbf{v}_1+\mathbf{v}_1\times\mathbf{v}_2)=(w_1w_2-\mathbf{v}_1\cdot\mathbf{v}_2, w_1\mathbf{v}_2+w_2\mathbf{v}_1+\mathbf{v}_1\times\mathbf{v}_2)$，**不可交换**，即 $q_1q_2\neq q_2q_1$。**共轭**：$q^*=(w,-\mathbf{v})$；**模长**：$\|q\|^2=w^2+\mathbf{v}^\top\mathbf{v}=qq^*=q^*q$；**逆**：$q^{-1}=\frac{q^*}{\|q\|^2}$

单位四元数 $\|q\|=1$，可表示三维空间中的旋转，$q^{-1}=q^*$。

旋转表示：绕某个单位向量 $\hat{\omega}$ 旋转 θ 角度，对应的四元数：$q=\left[\cos\frac{\theta}{2},\sin\frac{\theta}{2}\hat{\omega}\right]$

注意，旋转到四元数存在 **“双重覆盖”** 关系。从四元数恢复轴角表示：$\theta=2\arccos(w),\quad \hat{\omega}=\begin{cases}\frac{\mathbf{v}}{\sin(\theta/2)}, & \theta\neq 0\\ 0, & \theta=0\end{cases}$。

向量旋转：任意向量 $\mathbf{v}$ 沿着以**单位向量**定义的旋转轴 $\mathbf{u}$ 旋转 θ 度得到 $\mathbf{v}'$，那么：令向量 $\mathbf{v}$ 的四元数形式 $v=[0,\mathbf{v}]$，旋转四元数 $q=\left[\cos\left(\frac{\theta}{2}\right),\sin\left(\frac{\theta}{2}\right)\mathbf{u}\right]$，则旋转后的向量 $\mathbf{v}'$ 可表示为：$\mathbf{v}'=qvq^*=qvq^{-1}$。

如果是给定四元数 q 旋转向量 $\mathbf{v}$，那么设 $q=[w,\mathbf{r}]$ 是单位四元数（即 $w^2+\|\mathbf{r}\|^2=1$），向量 $\mathbf{v}$ 的四元数形式为 $v=[0,\mathbf{v}]$。

证明：倒三角函数，利用叉乘展开式：$a\times b\times c=(a\cdot c)b-(a\cdot b)c$ 和 $w^2+\|\mathbf{r}\|^2=1$ 即可。

旋转组合：两个旋转 q_1 和 q_2 的组合等价于四元数的乘法：$q_2(q_1xq_1^*)q_2^*=(q_2q_1)x(q_1^*q_2^*)$，不满足交换律，满足结合律。

令单位四元数 $q=w+x\mathbf{i}+y\mathbf{j}+z\mathbf{k}=[w,(x,y,z)]$，则旋转矩阵 $R(q)$ 为 $R(q)=\begin{bmatrix}1-2y^2-2z^2&2xy-2zw&2xz+2yw\\2xy+2zw&1-2x^2-2z^2&2yz-2xw\\2xz-2yw&2yz+2xw&1-2x^2-2y^2\end{bmatrix}$。根据上一步结果，旋转矩阵 R 的迹满足：$\mathrm{tr}(R)=3-4(x^2+y^2+z^2)=4w^2-1$，所以：$w=\frac{\sqrt{\mathrm{tr}(R)+1}}{2}$ $x=\frac{R_{32}-R_{23}}{4w}$ $y=\frac{R_{13}-R_{31}}{4w}$ $z=\frac{R_{21}-R_{12}}{4w}$。其中 R_{ij} 表示矩阵 R 的第 i 行第 j 列的元素。这些公式在 $w\neq 0$ 时有效。

在单位三维球面 S^3 上，或两个四元数 (q_1,q_2) 之间的角度 $\langle p,q\rangle=\arccos(p\cdot q)$。*证明*：设 $p=(p_w,\mathbf{p}_v)$ 和 $q=(q_w,\mathbf{q}_v)$，那么显然，从 p 旋转到 q 的相对旋转可以由四元数乘法 $\Delta q=qp^*$ 表示。$\Delta q=(q_wp_w+\mathbf{q}_v\cdot\mathbf{p}_v,\dots)$ 的实部是 $p\cdot q$。对应到旋转的距离就是乘个 2：$\mathrm{dist}(p,q)=2\min\{\langle p,q\rangle,\langle p,-q\rangle\}$。**要两个取最小值是因为双倍覆盖**。两个旋转 (R_1,R_2) 的距离与其对应四元数 $q(R_1)$ 和 $q(R_2)$ 在球面上的距离成线性关系（前者是后者的两倍）。

线性插值（Lerp）：$q(t)=(1-t)q_1+tq_2$。**归一化线性插值（Nlerp）**：$q(t)=\frac{(1-t)q_1+tq_2}{\|(1-t)q_1+tq_2\|}$（除个模长恢复为单位四元数）。以上两种插值都有问题，他们实际上是线性切分了弦长而不是弧长，会导致在转动时角速度不均匀。**球面线性插值（Slerp）**：$q(t)=\frac{\sin((1-t)\theta)}{\sin(\theta)}q_1+\frac{\sin(t\theta)}{\sin(\theta)}q_2$，其中 θ 是 q_1 和 q_2 之间的夹角，$\theta=\arccos(q_1\cdot q_2)$。*证明*：正弦定理。

球面均匀采样：：在 $\mathbb{SO}(3)$ 中均匀采样旋转矩阵等价于从单位四元数的集合 $\mathbb{S}(3)$ 中均匀采样。原因：两个旋转之间的距离与对应的四元数在单位球面上的距离成线性关系。均匀采样 $\mathbb{S}(3)$：从四维**标准正态分布** $\mathcal{N}(0,I_{4\times4})$ 中随机采样一个变量，并将其归一化，从而得到（直接解释为）单位四元数。原因：由于标准正态分布是各向同性的，所以采样得到的单位四元数在 $\mathbb{S}(3)$ 中也是均匀分布的。

- **旋转矩阵**：可逆、可组合（矩阵连乘）、但在 $\mathbb{SO}(3)$ 上移动不直接，**但最适合作为 NN 输出：连续性**。
- **欧拉角**：逆向复杂、组合复杂、因为 Gimbal lock 的存在，与 $\mathbb{SO}(3)$ 不能平滑映射
- **轴角**：可逆、组合复杂、大部分情况下可以与 $\mathbb{SO}(3)$ 平滑映射，但是在边界情况（如旋转 0 度时）不行
- **四元数**：可逆，可组合，平滑映射，但存在双倍覆盖的问题

碰撞检测：**球体包裹法（Bounding Spheres）**。缺点：**保守性导致可行解丢失**，限制了模型对于更精细物体的操作能力。（很小的面片，虚假自碰撞）

运动规划：**PRM 算法**：在 $\mathcal{C}_{\text{free}}$ 中随机 sample（不重），然后连接 k 近邻，剔除碰撞边（这一步用连线上线性采样来处理），然后用搜索或者 Dij 找路。**场景不变时可复用**。**高斯采样**：先随机 q_1，$\mathcal{N}(q_1,\sigma^2)$ 生成 q_2，如果 $q_1\in C_{\text{free}}$ 且 $q_2\notin C_{\text{free}}$，则添加 q_1。**有边界偏好，效率低**。**桥采样**：计算中点 q_3，当 q_1,q_2 都寄才要 q_3。*PRM 具有渐进最优性*。

RRT：exploration（随机采样）and exploitation（向着 goal 走）。探索参数 β、步长 ϵ 和采样点数量 n 都要调。**RRT-Connect**：双向 RRT、定向生长策略（扩展目标选择最近的另一树的叶子）、多种步长贪婪扩展。

Shortcutting：平滑路径。可以多次 RRT 后并行 shortcutting。

控制系统的性能评价：最小化稳态（Steady-State）误差；最小化调节时间，快速达到稳态；最小化稳态附近的振荡。**FeedForward 开环控制**：规划完闭眼做事无反馈纠错；**FeedBack 闭环控制**：带反馈回路，稳定。

期望状态：θ_d（destination），当前状态：θ，误差：$\theta_e=\theta_d-\theta$。

1. **稳态误差（Steady-State Error）**：系统到达稳态后的残余误差 $e_{ss}=\lim_{t\to\infty}\theta_e(t)$。理想系统应满足 $e_{ss}=0$
2. **调节时间（Settling Time）**：误差首次进入并保持在 $\pm 2\%$ 误差带所需时间
3. **超调量（Overshoot）**：系统响应超过稳态值的程度，最开始过去不算 overshoot $=|a/b|\times 100\%$，其中，a 表示最大偏移量，b 表示最终稳态值

P 控制 Proportional：$P=K_p\theta_e(t)$。**一阶形式**：当控制信号改变状态的导数（即控制速度信号）时：$\dot{\theta}(t)=P=K_p\theta_e(t)$，若希望状态以 $\dot{\theta}_d(t)=c$ 移动，则 $\dot{\theta}_e(t)+K_p\theta_e(t)=c$，解 ODE 得到 $\theta_e(t)=\frac{c}{K_p}+\left(\theta_e(0)-\frac{c}{K_p}\right)e^{-K_pt}$。当 $c\neq 0$（目标移动）时：随着 $t\to\infty$，$e^{-K_pt}\to 0$ 稳态误差：$\lim_{t\to\infty}\theta_e(t)=\frac{c}{K_p}$。**系统存在永久稳态误差**，误差大小与目标速度 c 成正比，与比例增益 K_p 成反比，增大 K_p 可以减小稳态误差。**二阶形式**：控制信号改变状态的二阶导数（力或力矩信号）：$\ddot{\theta}(t)=P=K_p\theta_e(t)$，导致状态振荡且不稳定。

PI 控制 Integral：$PI=K_p\theta_e(t)+K_i\int_0^t\theta_e(\tau)\mathrm{d}\tau$。如果控制信号作用于状态导数（速度）：$\dot{\theta}(t)=PI=K_p\theta_e(t)+K_i\int_0^t\theta_e(\tau)\mathrm{d}\tau$，$\dot{\theta}_e(t)=\dot{\theta}_d(t)-\dot{\theta}(t)$，$\dot{\theta}_d(t)=\dot{\theta}_e(t)+\dot{\theta}(t)$，两边求导：$\ddot{\theta}_d(t)=\ddot{\theta}_e(t)+K_p\dot{\theta}_e(t)+K_i\theta_e(t)$，如果 $\ddot{\theta}_d(t)=0$（目标加速度为零），$\ddot{\theta}_e(t)+K_p\dot{\theta}_e(t)+K_i\theta_e(t)=0$，为二阶常系数齐次微分方程。解的形式由方程特征根决定，特征方程为：$r^2+K_pr+K_i=0$。

其解的形式决定系统的阻尼特性：**过阻尼（Overdamped）**：两个实根，系统缓慢收敛。**临界阻尼（Critically damped）**：双重实根，快速无振荡收敛。**欠阻尼（Underdamped）**：共轭复根，系统振荡收敛。

PD 控制 Derivative：$PD=K_p\theta_e(t)+K_d\frac{\mathrm{d}}{\mathrm{d}t}\theta_e(t)$。如果 $\ddot{\theta}_d(t)=0$（目标加速度为零），则 $\ddot{\theta}_e(t)+K_d\dot{\theta}_e(t)+K_p\theta_e(t)=0$，解的形式由方程特征根决定，特征方程为：$r^2+K_dr+K_p=0$。

PID 控制：$PID=K_p\theta_e(t)+K_i\int_0^t\theta_e(\tau)\mathrm{d}\tau+K_d\frac{\mathrm{d}}{\mathrm{d}t}\theta_e(t)$。

K_p 控制当前状态：K_p 增大可 **加快响应速度（Rise Time）**、减少调节时间，因为快速减少 $\theta_e(t)$；**增大超调量**；**单用会产生稳态误差**。

K_i 控制历史累积：对持续误差进行累积补偿，**消除稳态误差**；**增大超调量**。

K_d 预测未来趋势：减少调节时间，**抑制超调和振荡**；当误差增加时提供更强的控制作用；当误差减小时提供更温和的控制作用。

2 Vision-and-Grasping

抓握式操作（Prehensile Manipulation）：通过完全约束物体自由度实现精确控制

非抓握式操作（Non-prehensile Manipulation）：利用推、滑等接触力学原理调整物体状态，适用于薄片状物体或预处理场景，**不是所有动作都需要抓取**

抓取姿势（Grasp Pose）：手的位置、方向和关节状态 & **抓取的自由度**

- **4-DoF 抓取**：仅需平移和绕重力轴旋转 (x,y,z,θ_z)
- **6-DoF 抓取**：允许任意方向接近 $(x,y,z,\theta_x,\theta_y,\theta_z)$
- **手指自由度**：平行夹爪：开 / 关，1 DoF；灵巧手（Dexterous Hand）：21 DoF

开环抓取系统：一般处理流程包括视觉感知、位姿估计、运动规划、控制执行，**闭环系统**：接收反馈信号重新抓取

对已知物体的抓取条件：预测物体位姿，RGB 图像需相机内参、物体大小（避免歧义 ambiguity）、**无对称性**；点云只需 **无对称性**

ICP：初始化变换估计 $T_0=(R_0,t_0)$，迭代（数据关联找变换后最近匹配点 + 变换求解 $T_{k+1}=\arg\min_T\sum_{(m,s)\in C}\|Tm-s\|^2$）直至收敛

对未知物体的抓取方法：直接预测抓取位姿，也有算法可以从见过同一类别物体进行泛化。

旋转回归：用神经网络估计连续的旋转变量，理想表达方式：表示空间到 $\mathbb{SO}(3)$ 群一一映射、且满足连续性，即不存在 **奇点（Singularities）**

欧拉角：$R=R_x(\alpha)R_y(\beta)R_z(\gamma)$，存在非双射（考虑引入冗余维度解决 $\theta\to(x,y)$）、Gimbal lock 问题

轴角：$(\text{axis},\text{angle})=(\mathbf{e},\theta)$，当 $\theta=0$（单位旋转）或 $\theta=\pi$（旋转轴可以反向）时存在多义性。

四元数：$q=w+xi+yj+zk$，存在双重覆盖问题，约束空间为上半球则引入不连续性（临近球大圆的不连续性和球大圆上的不连续性）。$q=\left[\cos\frac{\theta}{2},\sin\frac{\theta}{2}\hat{\omega}\right]$

6D 表示：拟合旋转矩阵然后施密特正交化（Schmidt orthogonalization），但拟合的 9 个数不等价

9D 表示：SVD 正交化，CNN 友好，连续、一一映射、等价。$\hat{R}=U\begin{bmatrix}1&0&0\\0&1&0\\0&0&\det(UV)\end{bmatrix}V^\top$（保证 $\det(\hat{R})=1$，符合旋转矩阵手性不变性）

增量旋转预测 Delta Rotation：小范围旋转，以上几种方式都适用，此时四元数等表示方法需要预测参数较少，学起来更快

Rotation Fitting：通过神经网络预测拟合物体表面点的对应关系（模型坐标系 (x_i^M,y_i^M,z_i^M) 到相机坐标系 (x_i^C,y_i^C,z_i^C) 的最优变换矩阵）。步骤：对物体表面的每个像素，预测其在物体建模模型上的 3D 坐标；基于这些对应关系拟合旋转矩阵（要求物体见过）

Orthogonal Procrustes：给定两组点 $\mathbf{M},\mathbf{N}\in\mathbb{R}^{n\times 3}$，求最优旋转矩阵 $\mathbf{A}$ 使得 $\hat{\mathbf{A}}=\arg\min_{\mathbf{A}}\|\mathbf{M}-\mathbf{N}\mathbf{A}^\top\|_F^2$，s.t. $\mathbf{A}^\top\mathbf{A}=\mathbf{I}$，其中 $\|X\|_F=\sqrt{\mathrm{trace}(X^\top X)}=\sqrt{\sum_{i,j}x_{ij}^2}$

解析解：对 $\mathbf{M}^\top\mathbf{N}$ 做 SVD 分解，$\mathbf{M}^\top\mathbf{N}=\mathbf{U}\mathbf{D}\mathbf{V}^\top$，最优旋转为 $\hat{\mathbf{A}}=\mathbf{U}\begin{bmatrix}1&0&0\\0&1&0\\0&0&\det(\mathbf{U}\mathbf{V}^\top)\end{bmatrix}\mathbf{V}^\top$（修正手性）

平移向量 t 可通过点集质心计算：$\hat{\mathbf{t}}=\overline{\mathbf{M}}-\hat{\mathbf{A}}\,\overline{\mathbf{N}}$

问题：对离群点敏感，常用 RANSAC 进行鲁棒拟合。

RANSAC：通过随机抽样找到内点（inliers）最多的模型假设，1. 以最小点集采样拟合所需模型（线 =2，面 =3）；2. 计算模型参数；3. 计算内点数量（距离小于阈值 ϵ 的）；4. 迭代重复，找内点最多的模型。

Instance Level PE：每个物体有独立模型，典型方法如 PoseCNN，结合 ICP 可提升精度

Category Level PE：同类物体归一化到 $1\times1\times1$ box，可预测旋转，预测平移需已知物体大小

迭代最近点算法（ICP）：作为后处理提高物体的位姿估计精度，提高抓取成功率；平移误差比旋转误差更敏感；怕物体被挡住造成 **点云缺失**。**目标**：优化初始位姿估计，对齐源点云 $P=\{p_i\}$ 和目标点云 $Q=\{q_j\}$；寻找最优旋转 $\hat{R}\in\mathbb{SO}(3)$ 和平移 $\hat{T}\in\mathbb{R}^{3\times1}$

流程：1. 中心化源、目标点集 $\tilde{p}_i=p_i-\bar{P},\tilde{q}_j=q_j-\bar{Q}$；2. **对应点匹配（Correspondence Search）**：为每个 $\tilde{p}_i$ 找到最近邻 $\tilde{q}_{j_i}$；3. 求解位姿（解见前）；4. 用解变换源点集 P，然后迭代 2-3 直至收敛。

ICP 收敛性：**不保证全局收敛**，可能陷入局部最优。原因：**对应点匹配可能非一一映射**，两个源点映到同一目标点。**优点**：简单，无需特征提取，初始估计好时精度高。**缺点**：计算成本高（最近邻搜索），对初始位姿敏感，迭代慢，未充分利用结构信息。

Category-Level Pose Estimation：解决实例级位姿估计需完整模型的问题，**通过归一化操作定义标准化物体空间 Normalized Object Coordinate Space（NOCS）**，包括旋转对齐、平移归一、尺寸归一。

ICP 算法需要很强的先验知识（物体的本身建模），然后进行变换前后点云配准，由于需要变换前后的**坐标对**，所以我们需要先进行**最近邻匹配**（也就是这一步导致了收敛性的缺失以及迭代速度的变慢），然后据此迭代得到物体位姿 (R,t)

NOCS 算法不需要完整的物体的本身建模，而是通过标准化的 NOCS 空间隐式地引入了对于某一类物体的、相较于 ICP 算法**更粗粒度**的几何先验，降低

了对于高精建模的依赖，使用**合成数据**训练得到一个神经网络，可以从 RGB 图像直接为每一个像素预测其在 NOCS 中的对应点 (x, y, z)，随后将其与 RGBD 重建得到的点云信息进行配准，**这里根据像素关系，可以天然形成数量相同的变换前后的坐标对，所以不再需要找到最近邻（Correspondence）**。而后，我们可以直接用 Umeyama 算法（和 ICP 去除最近邻匹配的后半段类似）来重建得到 7 DoF 物体位姿 (s, R, t)

1. 输入 RGBD 图像，提取 RGB 信息，使用 Mask R-CNN 获得 ROI（Region of Interest），分割物体
2. 预测每个像素的 NOCS 空间坐标 (x, y, z)，得到 **NOCS Map**
3. 将 NOCS Map 的点反投影（Back Projection）到三维空间中，得到点云数据 $\mathbf{q}_i$
4. 通过 NOCS Map 和 Depth 图像得到的点云数据，进行 Pose Fitting，利用 Umeyama 算法，计算得出物体的 7DoF 位姿（缩放 + 旋转 + 平移），缩放系数的计算就是简单的用 NOCS Map 的各轴向长度与物体实际点云各轴向作了一个除法。而反过来计算 Bounding Box 的时候，则利用了 NOCS 建模时令物体中心处在原点从而具有的对称性，以预测出的 NOCS Map 各轴向最大绝对值乘 2 再乘缩放系数作为了 Bounding Box 的各轴向尺寸

额外引入 NCOS 而不是直接 NN 预测原始点然后结合 Depth 直接回归 6DoF 位姿的原因：

1. 实验效果优
2. 将问题分解为 2D → 3D 映射 + 3D → 3D 几何优化，更直观
3. **NOCS 方法充分利用了形状 / 几何先验**，提升了对未见物体的泛化能力。

合成数据：训练 NOCS 网络需要大量标注数据，但真实数据标注成本高、泛化性差，所以需要合成数据进行训练。然而存在 **Sim2Real Gap**，导致模型在真实世界性能下降

Mixed Reality Data ：将合成前景物体叠加到真实背景上，易于获取大量带 NOCS 标签的数据。问题：合成前景与背景 **分界太过明显**，从而导致分割的 Mask R-CNN 学习到的经验难以应用到真实世界

Co-Training ：结合图像分割领域收集的真实数据集与合成数据集来一同对 Mask R-CNN 进行 **混合训练**，但前者不参与后续的 NOCS 映射训练，**只为分割提供监督信号**

后续处理：对于预测得到的位姿，有时候还需要 Refinement，比如之前介绍的 ICP 算法，也可通过神经网络（合成数据训练）完成

Form Closure：纯几何约束，不依赖摩擦力，最稳固

Force Closure：依赖摩擦力，通过接触力抵抗任意 Wrench（力 + 力矩），也即可以让物体产生的任何加速度 a 和角加速度 α

关系：Form Closure ⊆ Force Closure ⊆ Successful Grasp，反例：1. 双指夹纸；2. 托起。

摩擦锥（Friction Cone）：定义了在静摩擦系数 μ 下，接触点不滑动的力的方向范围（与法线夹角最大值 $\alpha = \arctan \mu$）

Force Closure 形式化：定义抓取矩阵（Grasp Matrix）$F = \begin{bmatrix} \mathcal{F}_1 & \cdots & \mathcal{F}_j \end{bmatrix} \in \mathbb{R}^{n \times j},\ n = 3 \text{ or } 6,\ j = k \times C$

其中，C 是接触点（摩擦锥）的数量，k 是为了近似每个摩擦锥所使用的力旋量数量（也即用多少面体锥来近似摩擦锥）。

1. 抓取矩阵必须满秩：$\text{rank}(F) = n$
2. 原点在凸锥内部（小正数下限保证稳定性）：$Fk = 0 \text{ for some } k \in \mathbb{R}^j, k_i \geq \epsilon > 0 \text{ for all } i$

GraspNet-1B：获得物体模型 → 在表面采样抓取位姿 → Force Closure 筛选 → 场景生成与物体位姿标注 → 变化抓取位姿到场景中，进行碰撞检测 → 多视角渲染扩增数据集。**意义**：证明了基于 3D 几何的模型能有效学习抓取，但大规模多样性需要合成数据。

摩擦系数 μ：GraspNet 数据集实际上为从低到高不同的 μ 值都进行了筛选并存储了标签。μ 值越低，对抓取的要求越高（更接近 Form Closure），这样的抓取在低摩擦表面上更可能成功。训练时可用小 μ，泛化性强。

Grasp Detection：从输入（点云 / TSDF / 图像）预测抓取位姿，包括位置、朝向、夹爪宽度、质量分数。

输入模态：往往 3D 几何通常优于 2D 信息，**因其直接包含形状信息**。

VGN (Volumetric Grasping Network) 输入：3D TSDF；架构：U-Net（3D CNN）；输出：预测三个体素网格，即**对于输出网格中的每一个体素**，网络预测 **抓取质量（Grasp Quality/Score）、抓取朝向（Grasp Orientation）、抓取宽度（Grasp Width）**（防止夹爪过宽向外碰撞）

特点：**依赖几何信息**、不依赖纹理、Sim2Real 效果好（优势在于 TSDF 对传感器噪声不敏感；现实物理效应（力闭合、变形）可能优于仿真；可以工程优化摩擦系数问题），**Sim2Real 甚至可以是负的！**

评估指标：通常是与物体无关、非任务导向的 **抓取成功率（Success Rate）、清理率（Percentage Cleard）、规划时间（Planning Time）**

后处理：**高斯平滑**（抓取评分非阶跃）、**距离掩膜**（限制夹爪在 TSDF 的区域）、**基于质量的 NMS**

VGN 的局限性：依赖多视角构建完整的场景 TSDF、精度受体素大小限制、高分辨率计算所需的内存成本高

GraspNet 成功的本质：

1. 点云的优良性质：准确、轻量、效率高、精度高
2. 架构：**端到端网络**，**多阶段设计**（先预测抓取分数，选高的做候选继续预测接近方向等），每个阶段都有监督信号，稳定
3. **泛化性**：局部性与平移等变性
   1. **局部性**：Cylinder Grouping 聚合，依赖候选点周围的局部几何信息判断，而不太关心场景中其他远处的物体。
   2. **平移等变性（Translation Equivariance）**：类似二维情形，模型学习到的几何模式识别能力不随物体在空间中的位置变化而失效。

Cylinder Grouping：在候选点附近，沿着接近方向定义一个圆柱体区域，聚合该区域内所有点的特征。这个聚合后的特征被用来预测最佳的旋转角和深度 / 宽度。

GraspNet 的核心在于 **学习局部几何特征（Local Geometric Features）与抓取成功的关系**，泛化性强

抓取的条件生成模型：替代检测方法，直接学习抓取位姿的分布（解决多峰分布）；尤其适用于高自由度灵巧手；常用条件扩散模型，基于局部几何特征生成抓取。

DexGraspNet：合成数据（Synthetic Data）+ 深度学习

1. 场景理解：预测每个点 **抓取可能性（Graspness）**，是否是 **物体（Objectness）**
2. 局部特征：不用全局特征（关联性弱、泛化性差），选择 Graspness 高的地方附近的点云，提取局部特征（**几何信息**）
3. 条件抓取生成模块：**条件生成处理** (T, R) **多峰分布**，然后采样后直接预测手指形态 θ

仅处理包覆式抓取（Power Grasp），没处理指尖抓取（Precision Grasp）；主要使用力封闭抓取；透明（Transparent）或高反光（Highly Specular/Shiny）物体有折射（Refraction）/ 镜面反射（Specular Reflection），导致点云质量差。

ASGrasp：深度修复，合成数据 + 监督学习。域随机化、多模态立体视觉、立体匹配（Stereo Matching）。

Affordance：指一个物体所能支持或提供的交互方式或操作可能性，哪个区域、何种方式进行交互。

Where2Act：大量随机尝试 + 标注。学习从视觉输入预测交互点 a_p、交互方向 $R_{z|p}$ 和成功置信度 $s_{R|p}$。**VAT-Mart**：预测一整条操作轨迹。

利用视觉输入进行预测：

- **物体位姿（Object Pose）**：需要模型、抓取标注。
- **抓取位姿（Grasp Pose）**：直接预测抓取点和姿态，无模型或预定义抓取。
- **可供性（Affordance）**

启发式（Heuristic）规则：预抓取 Pre-grasp，到附近安全位置再闭合，避免碰撞

1. **操作复杂度有限**：难以处理复杂任务，受启发式规则设计限制。
2. **开环执行（Open-loop）**：规划一次，执行到底，闭眼做事。高频重规划可近似闭环。

3 Policy

策略学习：学习 $\pi(a_t|s_t)$ 或 $\pi(a_t|o_t)$，实现 **闭环控制**。

BC：将 $D = \{(s_i, a_i^*)\}$ 视为监督学习任务，学习 $\pi_\theta(s) \approx a^*$。

Distribution shift：策略 π_θ 错误累积，访问训练数据中未见过的状态（$p_\pi(s)$ 与 $p_{\text{data}}(s)$ 不匹配），策略失效。

1. **改变 $p_{\text{data}}(o_t)$：Dataset Aggregation (DAgger)**

   训练 $\pi_i \Rightarrow$ 用 π_i **执行（Rollout）** 收集新状态 $\Rightarrow$ 查询专家在此状态下的 $a^* \Rightarrow D \leftarrow D \cup \{(s, a^*)\} \Rightarrow$ 重新训练 π_{i+1}。但是出错才标注，也会影响准确性。

2. **改变 $p_\pi(o_t)$（更好拟合）**：从（传统算法）最优解中获取；从教师策略中学习（有 **Privileged Knowledge** ）

遥操作数据（Teleoperation）：贵，也存在泛化问题。

非马尔可夫性：引入历史信息，但可能过拟合，因果混淆（Causal Confusion）。

目标条件化（Goal-conditioned）：$\pi(a|s, g)$，共享数据和知识。但 g 也有分布偏移问题。

IL 局限性：依赖专家数据、无法超越专家、不适用于需要精确反馈的高度动态 / 不稳定任务。

Offline Learning：固定数据集学习，无交互。

Online Learning：边交互边学习。

策略梯度定理：$\nabla_\theta J(\theta) \approx \frac{1}{N} \sum_{i=1}^N \nabla_\theta \log p_\theta(\tau^{(i)}) R(\tau^{(i)})$
$\nabla_\theta \log p_\theta(\tau) = \sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t|s_t)$ ，奖励函数无需可导。

环境模型：包括状态转移概率 $p(s_{t+1}|s_t, a_t)$ 和奖励函数 $r(s_t, a_t)$

- **Model-Free**：不需要知道环境的模型
- **Model-Based**：利用神经网络学习环境的模型

REINFORCE：按照整条轨迹的总回报 $R(\tau^{(i)})$ 加权，**On-Policy**。BC 是平权。

On-Policy：数据来自当前策略。效果好，**样本效率低**，每次都得重新采样。
Off-Policy：数据可来自不同策略。**样本效率高**，可能不稳定。

Reward-to-Go：降方差，用未来回报 $\hat{Q}(s_t, a_t) = \sum_{t'=t}^T r_{t'}$ 加权梯度。

Baseline：降方差，减去 a_t 无关状态基线 $b(s_t)$，$\hat{Q}(s_t, a_t) - b(s_t)$。梯度无偏。

Advantage $A^{\pi_\theta}(s_t, a_t) = Q^{\pi_\theta}(s_t, a_t) - V^{\pi_\theta}(s_t)$：动作相对平均的优势，可替换 $R(\tau^{(i)})$ 做权值，$\hat{A}(s_t, a_t) = r(s_t, a_t) + \gamma \hat{V}(s_{t+1}) - \hat{V}(s_t)$

Bootstrap：使用基于当前函数估计的值 $\hat{V}_\phi^\pi(s_{i,t+1})$ 来更新 **同一个函数** 在另一个点 $s_{i,t}$ 的估计 $\hat{V}_\phi^\pi(s_{i,t})$

Batch AC：收集一批完整轨迹或转换数据后，统一更新 A/C。梯度估计更稳定，但更新频率低。

Online AC：每一步交互（或极小批量）后，立即更新 A/C。更新快，数据利用率高，但梯度估计方差较大。

Parallelization：多 worker 采样，提速增稳，异步快。