UC San Diego

L11: 6D Pose Estimation (Cont')

Hao Su

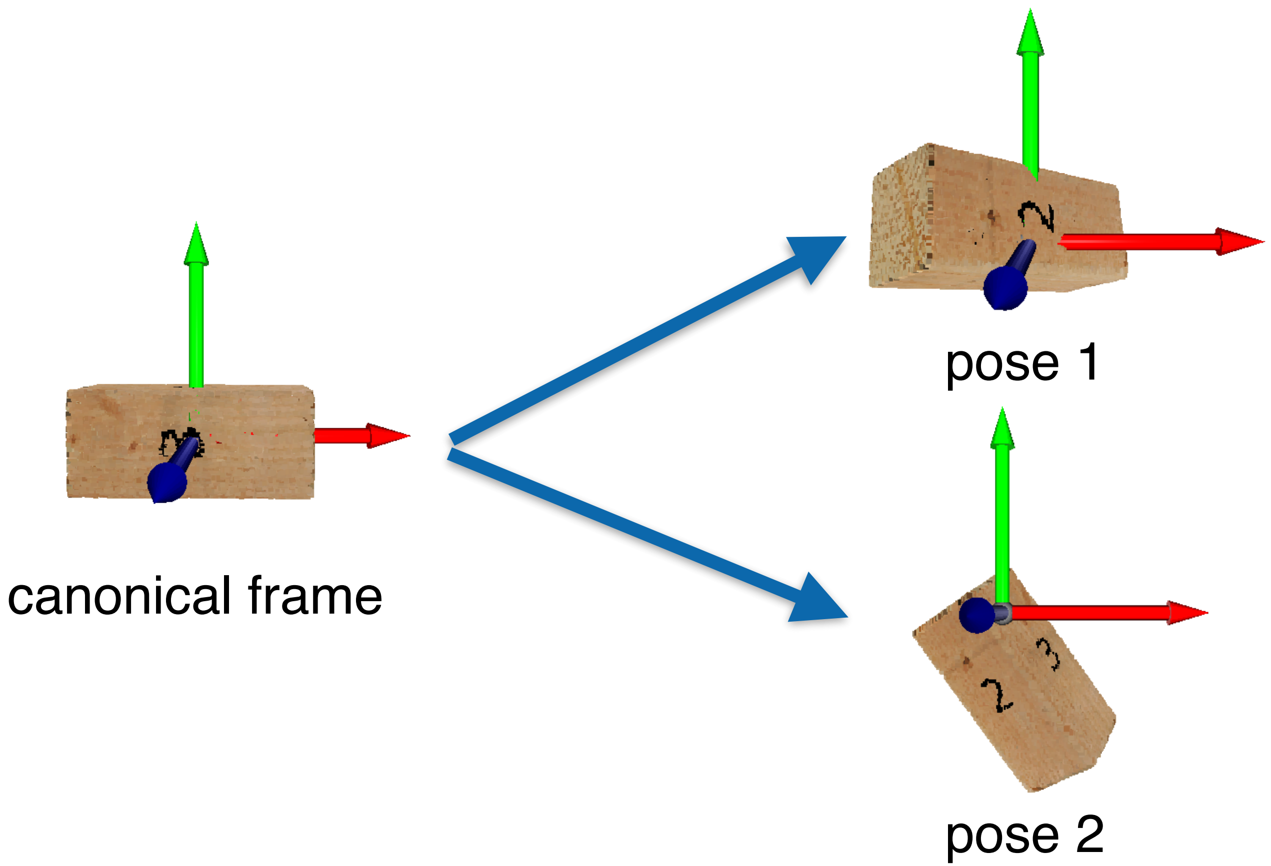
Ack: Minghua Liu and Jiayuan Gu for helping to prepare slides

We are talking about object pose

Figure from https://paperswithcode.com/task/6d-pose-estimation

Review

6D Pose Estimation



recognize the 3D location and orientation of an object relative to a canonical frame

Review

Agenda

- Introduction
- Rigid transformation estimation
  - Closed-form solution given correspondences
  - Iterative closet point (ICP)
- Learning-based approaches
  - Direct approaches
  - Indirect approaches

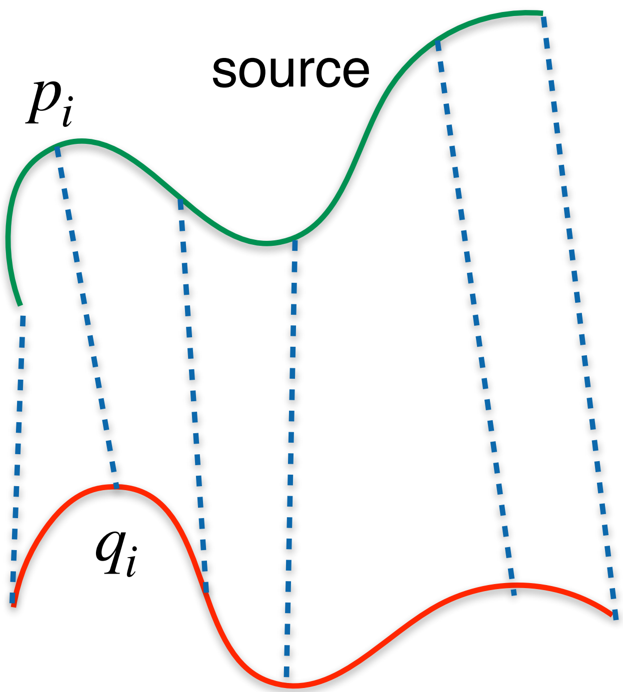
Review

Rigid Transformation Estimation

Review

Correspondence

Rigid transformation $T(p) = Rp + t$



$$q_1 = T(p_1) = Rp_1 + t$$

$$q_2 = T(p_2) = Rp_2 + t$$

$$\vdots$$

$$q_n = T(p_n) = Rp_n + t$$

3n equations from
n pairs of points

Review

Umeyama's Algorithm

- Known: $P = \{p_i\}, Q = \{q_i\}$
- Objective: $\min_{R,t} \sum_{i=1}^{n} \|Rp_i + t - q_i\|^2$
- Solution
    - $\sum_{i=1}^{n} (q_i - \bar{q}_i)(p_i - \bar{p}_i)^T = U\Sigma V^T$ (SVD)
    - $R = UV^T$ (flip the sign of the last column of V if $\det(R) = -1$)
    - $t = \frac{\sum_{i=1}^{n} q_i}{n} - \frac{\sum_{i=1}^{n} Rp_i}{n} := \bar{q} - R\bar{p}$

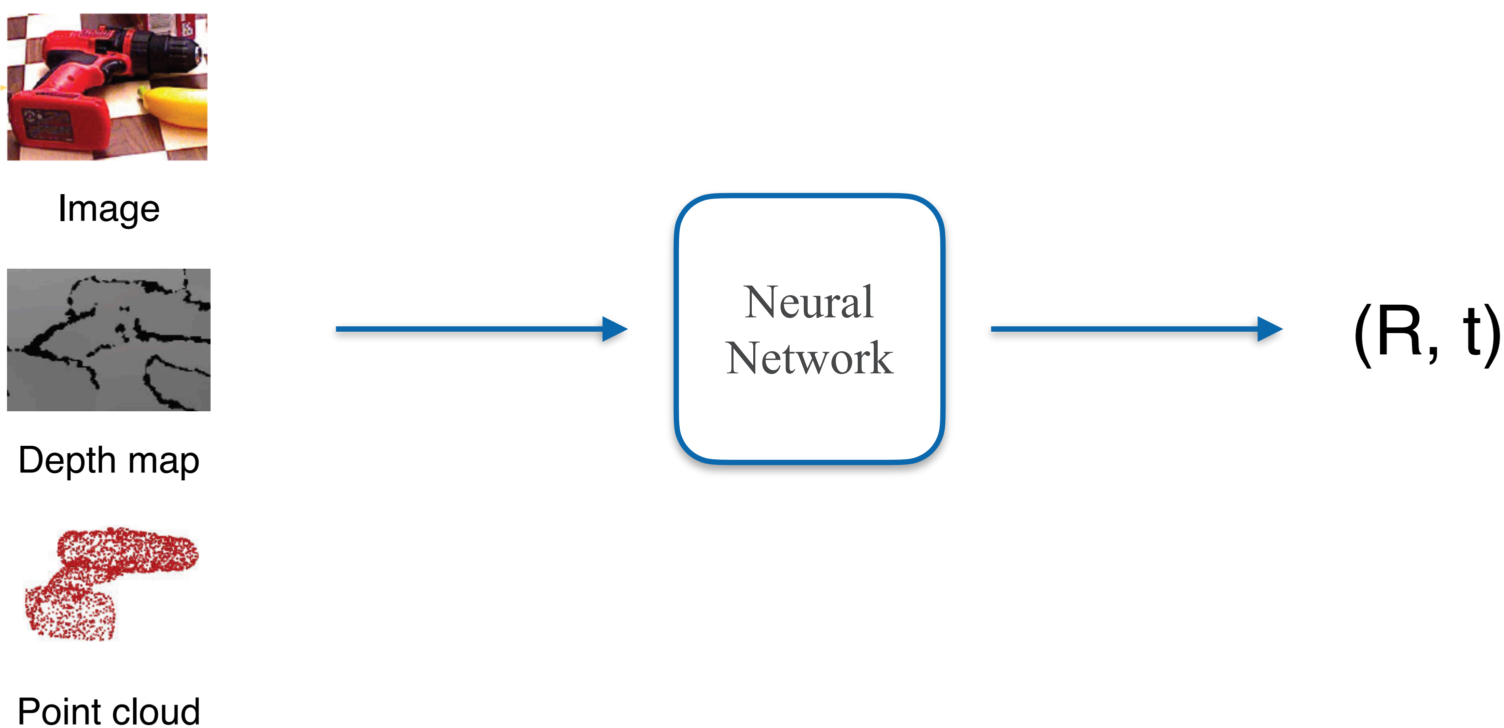
Review

Direct Approaches

Review

Direct Approaches

- Input: cropped image/point cloud/depth map of a single object
- Output: (R, t)



Review

Indirect Approaches

Indirect Approaches

- Input: cropped image/point cloud/depth map of a single object
- Output: corresponding pairs $\{(p_i, q_i)\}$
  - points in canonical frame $\{p_i\}$
  - points in camera frame $\{q_i\}$
  - estimate (R, t) by solving

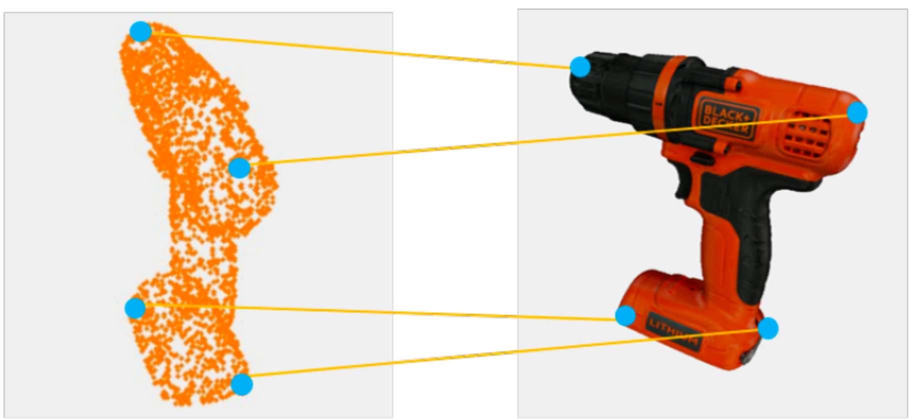
$$\min_{R,t} \sum_{i=1}^{n} \|Rp_i + t - q_i\|^2$$

Two Categories of Indirect Approaches

- If points in the canonical frame are known, predict their corresponding locations in the camera frame
- If points in the camera frame are known, predict their corresponding locations in the canonical frame

Given Points in the Canonical Frame, Predict Corresponding Location in the Camera Frame

- Recall: Three correspondences are enough
- Which points in the canonical frame should be given?
    - Choice by PVN3D: keypoints in the canonical frame



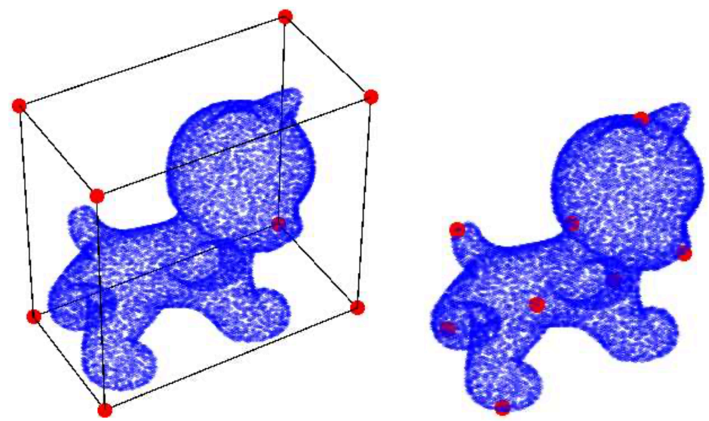
(d) 3D keypoints (cam.) (e) 3D keypoints (obj.)

He et al., "PVN3D: A Deep Point-wise 3D Keypoints Voting Network for 6DoF Pose Estimation", CVPR 2020

Keypoint Selection

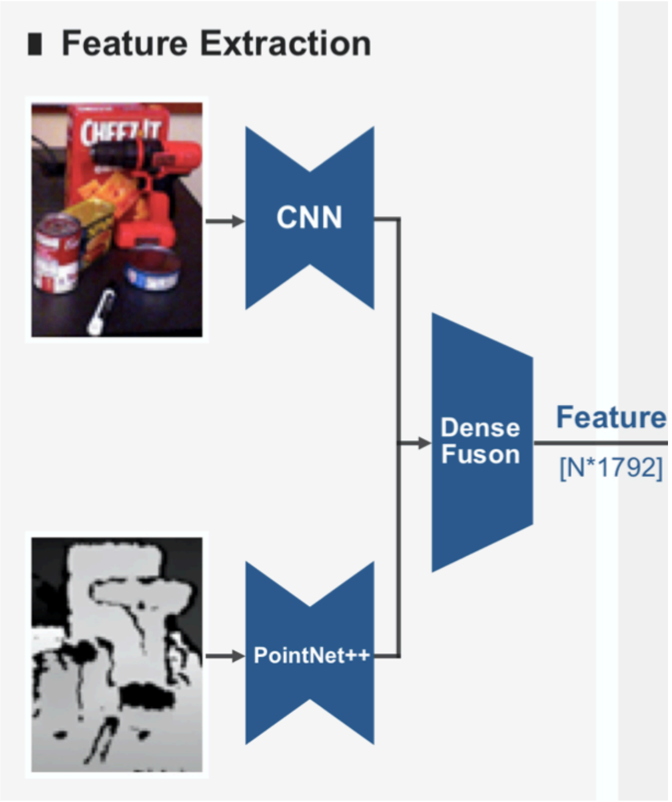
- Option1: bounding box vertices

- Option 2: farthest point sampling (FPS) over CAD object model



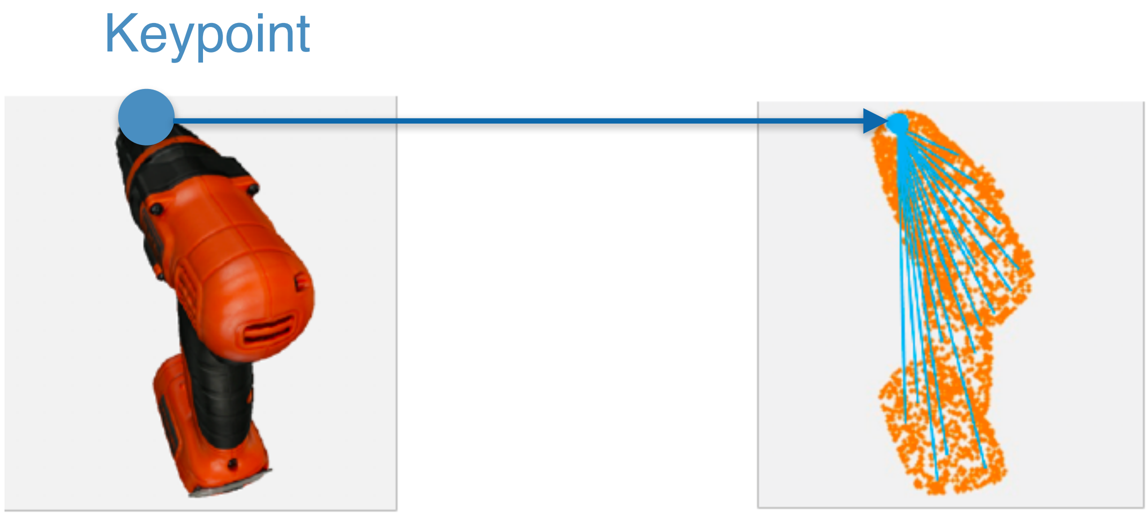
Chen et al., "G2L-Net: Global to Local Network for Real-time 6D Pose Estimation with Embedding Vector Features", CVPR 2020

Example: PVN3D



Get point-wise features
by fusing color and geometry features

Example: PVN3D



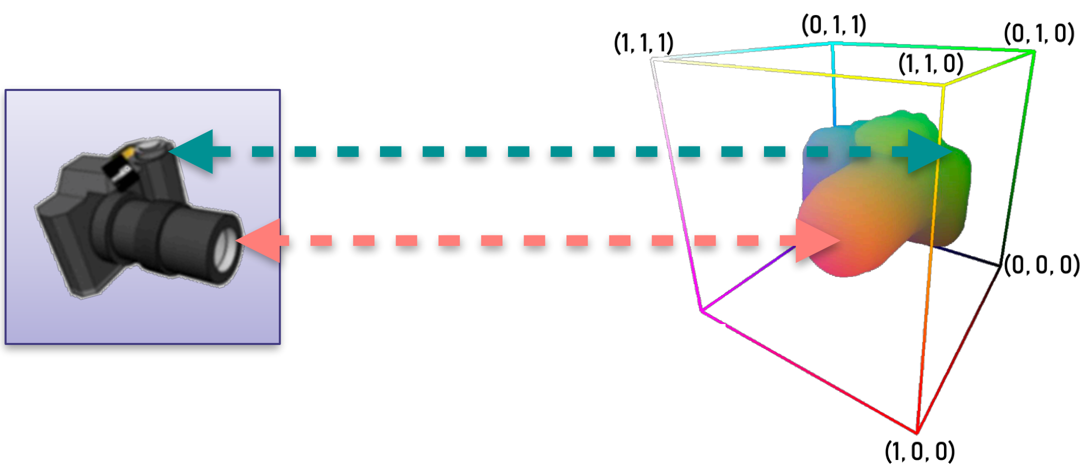
(c) Voting & clustering

For each keypoint:

- **Voting**: for each point in the camera frame, predict its offset to the keypoint (in the camera frame)
- **Clustering**: find one location according to all the candidates

Given Points in the Camera Frame, Predict Corresponding Location in the Canonical Frame

- Which points in the camera frame should be given?
  - Choice by NOCS: every point in the camera frame



3D point in the camera frame (2D visible pixel with depth)

3D point in the (normalized) canonical frame

Wang, He, et al. "Normalized object coordinate space for category-level 6d object pose and size estimation." CVPR 2019.

Example: NOCS

Input Image

Output NOCS Map
$H \times W \times 3$

NOCS Map

Note: the object is normalized to have unit diagonal of bounding box in the canonical space, so the canonical space is called "Normalized Object Canonical Space" (NOCS)

Wang, He, et al. "Normalized object coordinate space for category-level 6d object pose and size estimation." CVPR 2019.

Example: NOCS for Symmetric Objects

- Given equivalent GT rotations $\mathscr{R} = \{R_{GT}^{1}, R_{GT}^{2}, \cdots, R_{GT}^{n}\}$ (finite symmetry order n), we can generate n equivalent NOCS maps
- Similar to shape-agnostic loss in direct approaches, we can use *Min of N* loss

Wang, He, et al. "Normalized object coordinate space for category-level 6d object pose and size estimation." CVPR 2019.

Umeyama's Algorithm with Unknown Scale

- However, the target points in the canonical space of NOCS are normalized, and thus we also need to predict the scale factor

- Similarity transformation estimation (rigid transformation + uniform scale factor)

- Closed-form solution
  - Umeyama algorithm: http://web.stanford.edu/class/cs273/refs/umeyama.pdf
  - Similar to the counterpart without scale

Read by yourself

Tips for Homework 2

- For learning-based approaches
  - Start with direct approaches
  - Crop the point cloud of each object from GT depth map given GT segmentation mask
  - Train a neural network, e.g. PointNet, with shape-agnostic loss
  - Improve the results considering symmetry