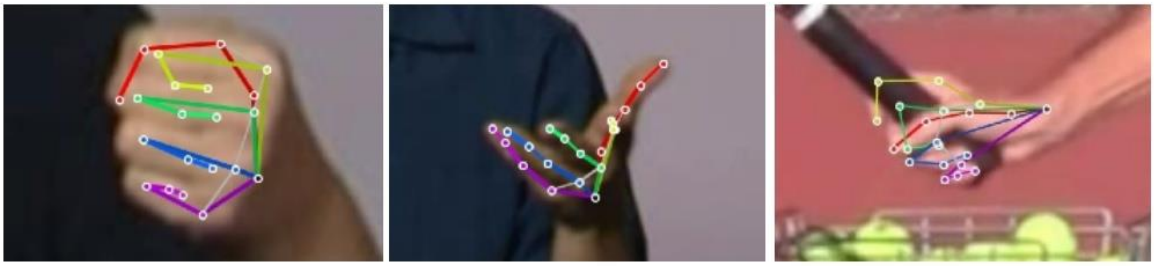
0. paper

Hand Keypoint Detection in Single Images using Multiview Bootstrapping

1. motivation

Due to heavy occlusions, even manual hand keypoint annotations are difficult to get right.



(a) Articulation (b) Viewpoint (c) Object

Figure 2: Hand annotation is difficult in single images because joints are often occluded due to (a) articulations of other parts of the hand, (b) a particular viewing angle, or (c) objects that the hand is grasping.

based on the observation: even if a particular image of the hand has significant occlusion, there often exists an **unoccluded view.**

2. multiview bootstrapped training

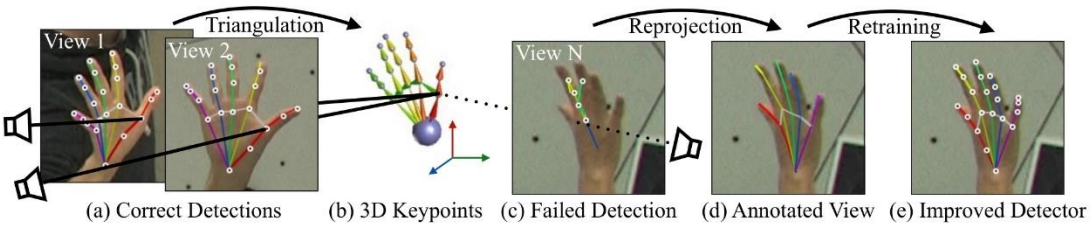


Figure 3: Multiview Bootstrapping. (a) A multiview system provides views of the hand where keypoint detection is easy, which are used to triangulate (b) the 3D position of the keypoints. Difficult views with [illegible] detections can be (c) annotated using the reprojected 3D keypoints, and used to retrain (e) an improved detector that now works on difficult views.

Given the **initial keypoint detector d0** and a dataset of **unlabeled multiview images**, our objective is to use the detector to generate a set of labeled images, which can be used to train an improved detector d1.

$$d(\mathbf{I}) \mapsto \{(\mathbf{x}_p, c_p) \text{ for } p \in [1 \dots P]\}\,. \tag{1}$$

detector d(**I**);

Image: **I;**

P: hand keypoint number;

xp: keypoint **2d location**;

cp: **detection confidence** of xp.

Training set with annotations:

$$\mathcal{T}_0 := \left\{ \left(\mathbf{I}^f, \{\mathbf{y}_p^f\} \right) \text{ for } f \in [1 \dots N_0] \right\}, \tag{2}$$

for frame f, y is the labeled keypoint, total N0 pairs.

Algorithm 1 Multiview Bootstrapping

Inputs:

- Unlabeled images: $\{\mathbf{I}_v^f$ for $v \in$ views, $f \in$ frames$\}$
- Keypoint detector: $d_0(\mathbf{I}) \mapsto \{(\mathbf{x}_p, c_p)$ for $p \in$ points$\}$
- Labeled training data: $\mathcal{T}_0$

for iteration i in 0 to K:

1. Triangulate keypoints from weak detections
   for every frame f:
   (a) Run detector $d_i(\mathbf{I}_v^f)$ on all views v (Eq. (5))
   (b) Robustly triangulate keypoints (Eq. (6))
2. Score and sort triangulated frames (Eq. (7))
3. Retrain with N-best reprojections (Eq. (8))
   $d_{i+1} \leftarrow \text{train}(\mathcal{T}_0 \cup \mathcal{T}_{i+1})$

Outputs: Improved detector $d_K(\cdot)$ and training set $\mathcal{T}_K$

Given **V** views of an object in a particular frame **f**:

$$\mathbf{X}_p^f = \arg\min_{\mathbf{X}} \sum_{v \in \mathcal{I}_p^f} ||\mathcal{P}_v(\mathbf{X}) - \mathbf{x}_p^v||_2^2, \qquad (6)$$

Try to **minimize the reprojection error** to obtain the final triangulated position.

The triangulation is RANSAC approach.

Then order the frames by P keypoints` score:

$$\text{score}(\{\mathbf{X}_p^f\}) = \sum_{p \in [1 \ldots P]} \sum_{v \in \mathcal{I}_p^f} c_p^v. \tag{7}$$

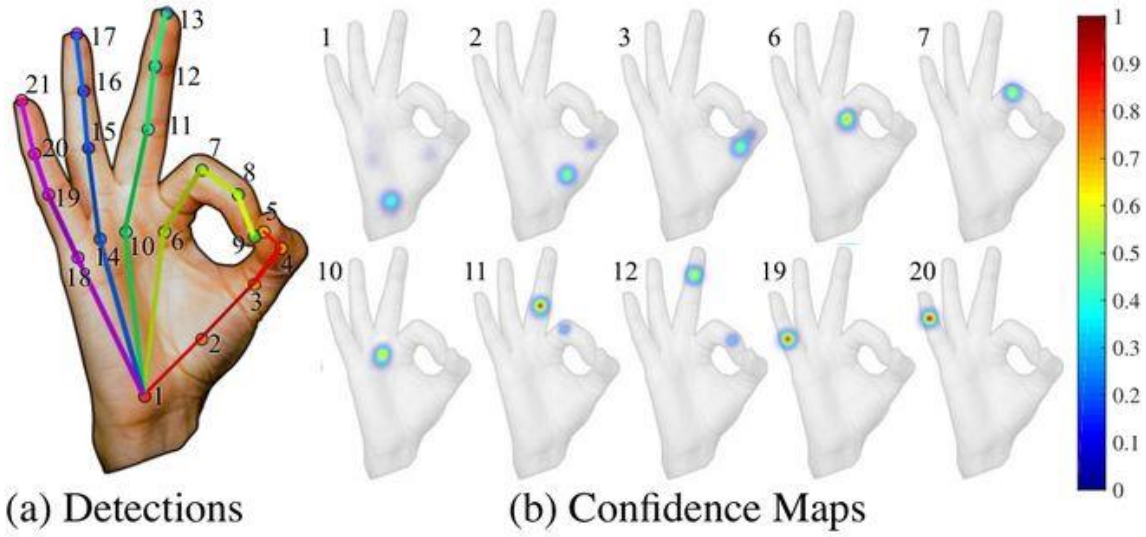


Figure 4: (a) Input image with 21 detected keypoints. (b) Selected confidence maps produced by our detector, visualized as a "jet" colormap overlaid on the input.

finally, Retraining with N-best Reprojections.