Normalization of RGB hand dataset RHD

Overview of dataset

- Dataset split
  1. Train: 41258
  2. Test: 2728
- Each image is composed of
  1. Left hand (21 joints)
  2. Right hand (21 joints)
- Ground Truth Annotation
  1. 3D (unit: probably m?)
  2. Camera Parameters
  3. 2D projection
  4. Visible mask

Editor - prepare_test_predictions_mat.m

frame1005.xyz | frame0 | frame0.xyz

frame0.xyz

	1	2	3	4
1	-0.0049	-0.0572	0.7018	
2	-0.0141	0.0303	0.5829	
3	-0.0206	0.0136	0.6067	
4	-0.0283	-0.0134	0.6344	
5	-0.0204	-0.0345	0.6679	
6	0.0198	0.0278	0.5292	
7	0.0143	0.0153	0.5475	
8	0.0077	-0.0034	0.5729	
9	-0.0016	-0.0203	0.6021	
10	0.0236	0.0625	0.5643	
11	0.0257	0.0351	0.5711	
12	0.0268	0.0088	0.5783	
13	0.0205	-0.0173	0.6071	
14	0.0355	0.0586	0.5791	
15	0.0378	0.0308	0.5893	
16	0.0392	0.0078	0.5961	
17	0.0346	-0.0155	0.6197	

frame0 | frame0.xyz | fr

frame0.uv_vis

	1	2	3
1	158	136.9000	1
2	153.2000	174.7000	1
3	150.4000	166.3000	1
4	147.4000	154	1
5	151.4000	145.4000	1
6	170.6000	174.9000	1
7	167.4000	167.9000	1
8	163.8000	158.3000	1
9	159.2000	150.5000	1
10	171.8000	191.3000	1
11	172.7000	177.4000	1
12	173.1000	164.3000	1
13	169.6000	151.9000	1
14	177.3000	188.6000	1
15	178.2000	174.8000	1
16	178.6000	163.7000	1
17	175.8000	152.9000	1

Editor - prepare_test_predictions_mat.m

frame1005.xyz | frame0 | frame0.xyz | frame0.K

frame0.K

	1	2	3	4	5	6
1	282.9000	0	160			
2	0	282.9000	160			
3	0	0	1			
4						
5						
6						

Overview of dataset

- Camera Parameters

$$K = \begin{matrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{matrix}$$

Where f_x is focus length (x direction), f_y is focus length (y direction)
In the dataset, as far as I am concerned, f_x is equal f_y

u_0 is x direction offset of raw image, and v_0 is y direction offset

$$KP = proj$$

$$\begin{matrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{matrix} \times \begin{matrix} \frac{x_k}{z_k} \\ \frac{y_k}{z_k} \\ 1 \end{matrix} = \begin{matrix} f_x \frac{x_k}{z_k} + u_0 \\ f_y \frac{y_k}{z_k} + v_0 \end{matrix} = \begin{matrix} u_k \\ v_k \end{matrix}$$ (for instance, joint k)

2.5D Pose Representation

- × Predict absolute 3D in *camera coordinate* is infeasible given the projection and scale ambiguity.
- *Instead,* predict 2.5D pose

$$P_k^{2.5D} = (u_k, v_k, Z_k^r)$$

Where u_k, v_k are image coordinates

Z_k^r is root-relative depth value (relative location to root joint)

2.5D Pose Representation

- One thing to mention is that different hand has different scales
  Remove the scale ambiguity by

Scale Normalization *(P is 3D before scale normalization)*

$$\hat{P} = \frac{C}{s} P$$

C is a constant set to 1.0, $s = \left\| P_n - P_{parent(n)} \right\|_2$ is the bone length of a specific bone (palm root -> index MCP here)

Thus the bone length after normalization is

$\hat{s} = \left\| \widehat{P_n} - \widehat{P_{parent(n)}} \right\|_2$ which should be the constant C

2.5D Pose Representation

- Look at the formula

$$KP = proj$$
$$K\hat{P} = \widehat{proj} = proj$$

The scale-normalized 2.5D pose we want to predict is

$$P_k^{2.5D} = \left(\widehat{u_k}, \widehat{v_k}, \widehat{Z_k^r}\right) = (u_k, v_k, \widehat{Z_k^r})$$

About u_k, v_k (projection on raw image)

➢ the network predicts local projection normalized in [0, 1], which can be recovered back to global location provided with bounding box crop information

2.5D Pose Representation

About $\widehat{Z_k^r}$

- The paper says that all hand centers are approximately in a range between 40 cm and 65 cm,
- The real values of each joint's depth lies approximately in a range 0.40 to 0.65,
- Thus one can analyze the training dataset to crop a rough cube, for each sample, which contains the hand propitiously (the cube size in optical axis direction **should be a constant value**)
- And then normalize the "scale-normalized root-relative" depth in that range

Overall Architecture

Infer 2.5D from CNN

- Scale-normalized root-relative joint prediction get

Scale normalized root Z

- Depth value of scale-normalized root $\widehat{Z_{root}}$ get

Scale recovery

- Recover the scale-normalized absolute joint prediction to absolute w/o scale norm.

Scale-normalized root-relative joint

- Input: cropped monocular RGB hand image I
- Output: scale-normalized root-relative joint in 2.5D space $\widehat{P^{2.5D}} = \left\{\left(u_k, v_k, \widehat{Z_k^r}\right)\right\}_{k \in K}$

Methodology: CNN

$$f(I) = \widehat{P^{2.5D}}$$

✓Regression

✓Heatmap (2D/3D)

✓Other map representation (fully conv network)

Scale-normalized depth value of root

- Estimate $\widehat{Z_{root}}$

Let n be index MCP, m be parent of n : palm root

If intrinsic camera parameter is unknown, multiple 3D solutions can have same 2D projection

Given camera parameters f_x, f_y,u_0, v_0, one unique solution:

$$\left(\widehat{X_n} - \widehat{X_m}\right)^2 + \left(\widehat{Y_n} - \widehat{Y_m}\right)^2 + \left(\widehat{Z_n} - \widehat{Z_m}\right)^2 = C^2$$

(remember the aforementioned **scale normalization** step)

Scale-normalized depth value of root

- $$\begin{matrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{matrix} \times \begin{matrix} \frac{x_k}{z_k} \\ \frac{y_k}{z_k} \\ 1 \end{matrix} = \begin{matrix} f_x \frac{x_k}{z_k} + u_0 \\ f_y \frac{y_k}{z_k} + v_0 \end{matrix} = \begin{matrix} u_k \\ v_k \end{matrix}$$

$\widehat{x_n} = \frac{(u_n - u_0)\widehat{z_n^r}}{f_x}, \widehat{x_m} = \frac{(u_m - u_0)\widehat{z_m^r}}{f_x}, \widehat{y_n} = \frac{(v_n - v_0)\widehat{z_n^r}}{f_y}, \widehat{y_m} = \frac{(v_m - v_0)\widehat{z_m^r}}{f_y}$

Let $\widehat{z_{root}} = t$

Scale-normalized depth value of root

- $\left(\frac{u_n-u_0}{f_x}\left(t+\widehat{z_n^r}\right)-\frac{u_m-u_0}{f_x}\left(t+\widehat{z_m^r}\right)\right)^2+$ $\left(\widehat{X_n}-\widehat{X_m}\right)^2$
- $\left(\frac{v_n-v_0}{f_y}\left(t+\widehat{z_n^r}\right)-\frac{v_m-v_0}{f_y}\left(t+\widehat{z_m^r}\right)\right)^2+$ $\left(\widehat{Y_n}-\widehat{Y_m}\right)^2$
- $\left(\left(t+\widehat{z_n^r}\right)-\left(t+\widehat{z_m^r}\right)\right)^2-C^2=0$ $\left(\widehat{Z_n}-\widehat{Z_m}\right)^2-C^2$

Which means

Scale-normalized depth value of root

- $\left(\left(\frac{u_n-u_m}{f_x}\right)t+\frac{u_n-u_0}{f_x}\widehat{Z_n^r}-\frac{u_m-u_0}{f_x}\widehat{Z_m^r}\right)^2+$
- $\left(\left(\frac{v_n-v_m}{f_y}\right)t+\frac{v_n-v_0}{f_y}\widehat{Z_n^r}-\frac{v_m-v_0}{f_y}\widehat{Z_m^r}\right)^2+$
- $\left(\widehat{Z_n^r}-\widehat{Z_m^r}\right)^2-C^2=0$

To quadratic formula

Scale-normalized depth value of root

- $\left(\frac{u_n - u_m}{f_x}\right)^2 t^2 + \left(\frac{v_n - v_m}{f_y}\right)^2 t^2 + 2t\left(\left(\frac{u_n - u_m}{f_x}\right)\left(\frac{u_n - u_0}{f_x}\widehat{Z_n^r} -\right.\right.$

Scale-normalized depth value of root

- Let r_A be $\left(\frac{u_n - u_m}{f_x}\right)^2 + \left(\frac{v_n - v_m}{f_y}\right)^2$

r_B be $2(\left(\frac{u_n - u_m}{f_x}\right)\left(\frac{u_n - u_0}{f_x}\widehat{Z_n^r} - \frac{u_m - u_0}{f_x}\widehat{Z_n^r}\right) + \left(\frac{V_n - V_m}{f_y}\right)\left(\frac{v_n - v_0}{f_y}\widehat{Z_n^r} -\right.$

Scale-normalized depth value of root

- Then $r\_A\ t^2 + r\_B\ t + r\_C\ = 0$

$$t = \frac{-r\_B \pm \sqrt{r\_B^2 - 4r\_A\ r\_C}}{2r\_A}$$

Only one t is valid ($bigger\ one$)

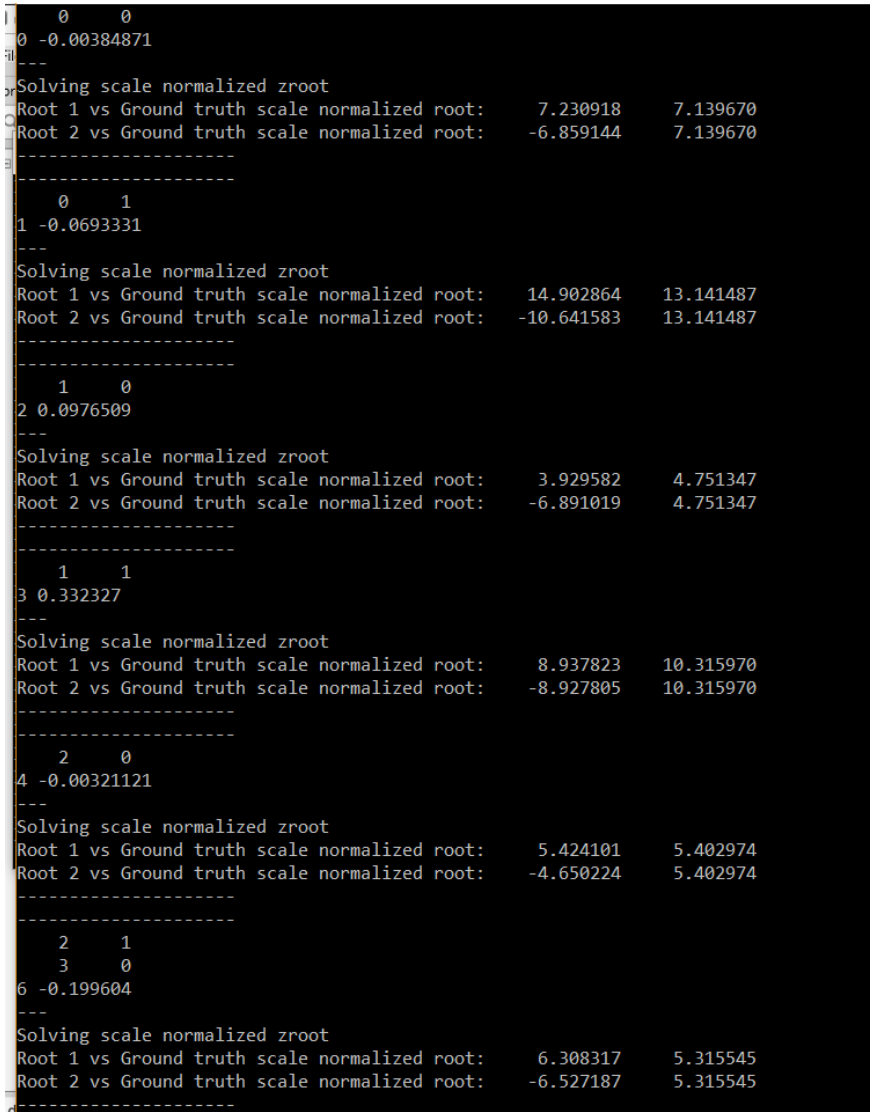
Scale-normalized depth value of root

- Screenshot of code
  Validated on several training samples using the

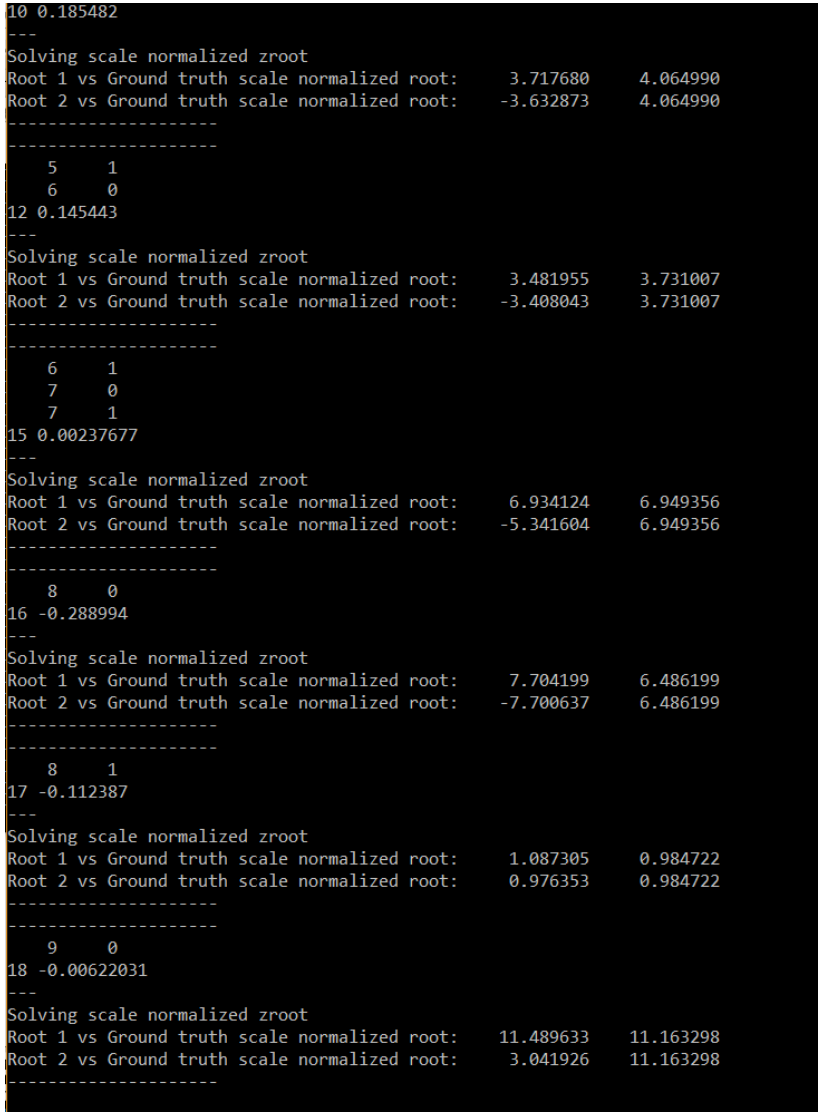
1. Ground truth 2D projection
2. Ground truth 3D (for scale norm calculation)
3. Camera parameters

To solve the scale-normalized depth value of root $\widehat{Z_{root}}$

Scale-normalized depth value of root



```
   0     0
0 -0.00384871
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:     7.230918     7.139670
Root 2 vs Ground truth scale normalized root:    -6.859144     7.139670
---------------------
---------------------
   0     1
1 -0.0693331
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:    14.902864    13.141487
Root 2 vs Ground truth scale normalized root:   -10.641583    13.141487
---------------------
---------------------
   1     0
2 0.0976509
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:     3.929582     4.751347
Root 2 vs Ground truth scale normalized root:    -6.891019     4.751347
---------------------
---------------------
   1     1
3 0.332327
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:     8.937823    10.315970
Root 2 vs Ground truth scale normalized root:    -8.927805    10.315970
---------------------
---------------------
   2     0
4 -0.00321121
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:     5.424101     5.402974
Root 2 vs Ground truth scale normalized root:    -4.650224     5.402974
---------------------
---------------------
   2     1
   3     0
6 -0.199604
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:     6.308317     5.315545
Root 2 vs Ground truth scale normalized root:    -6.527187     5.315545
---------------------
```



```
10 0.185482
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:     3.717680     4.064990
Root 2 vs Ground truth scale normalized root:    -3.632873     4.064990
---------------------
---------------------
   5     1
   6     0
12 0.145443
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:     3.481955     3.731007
Root 2 vs Ground truth scale normalized root:    -3.408043     3.731007
---------------------
---------------------
   6     1
   7     0
   7     1
15 0.00237677
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:     6.934124     6.949356
Root 2 vs Ground truth scale normalized root:    -5.341604     6.949356
---------------------
---------------------
   8     0
16 -0.288994
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:     7.704199     6.486199
Root 2 vs Ground truth scale normalized root:    -7.700637     6.486199
---------------------
---------------------
   8     1
17 -0.112387
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:     1.087305     0.984722
Root 2 vs Ground truth scale normalized root:     0.976353     0.984722
---------------------
---------------------
   9     0
18 -0.00622031
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:    11.489633    11.163298
Root 2 vs Ground truth scale normalized root:     3.041926    11.163298
---------------------
```

Scale recovery

- Need to know the global hand scale

The big idea is that when estimated scale normalized root-relative 3D is scaled, the difference from mean bone length (averaged over all training samples) is minimized. (For each bone)

Formula:

$$\hat{s} = \underset{s}{argmin} \sum_{k,l \in \varepsilon} \left(s \cdot \left\| \widehat{P_k} - \widehat{P_l} \right\|_2 - \mu_{kl} \right)^2$$

Where μ_{kl} is average bone length between keypoint k and keypoint l on training set.

Scale recovery

- $\sum_{k,l\in\varepsilon}\left(scale\left\|\widehat{P_k}-\widehat{P_l}\right\|_2-\mu_{kl}\right)^2=$

$$scale^2\sum_{k,l\in\varepsilon}\left(\left\|\widehat{P_k}-\widehat{P_l}\right\|_2\right)^2+scale\cdot(-2)\cdot\sum_{k,l\in\varepsilon}(\left\|\widehat{P_k}-\widehat{P_l}\right\|_2\cdot\mu_{kl})+$$

$\sum_{k,l\in\varepsilon}(\mu_{kl})^2=$

$$s\_A\ scale^2+s\_B\ scale+s\_C$$

The equation reaches its nadir at $scale=\frac{s\_B}{-2\ s\_A}$

Scale recovery

- Where

$$s\_A = \sum_{k,l\in\varepsilon}\left(\left\|\widehat{P_k} - \widehat{P_l}\right\|_2\right)^2,$$

$$s\_B = (-2)\cdot\sum_{k,l\in\varepsilon}(\left\|\widehat{P_k} - \widehat{P_l}\right\|_2 \cdot \mu_{kl}),$$

$$s\_C = \sum_{k,l\in\varepsilon}(\mu_{kl})^2$$

Quite elegant!

Scale recovery

```
Root 2 vs Ground truth scale normalized root:    -4.669377     4.840677
---------------------
Solving global hand scale which is
Solved global scale vs real global scale:     0.110433     0.126036
---------------------
   10     1
21 -0.112877
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:     5.978505     5.392392
Root 2 vs Ground truth scale normalized root:    -4.817154     5.392392
---------------------
Solving global hand scale which is
Solved global scale vs real global scale:     0.111744     0.123711
---------------------
   11     0
22 -0.122755
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:    10.824253     9.879951
Root 2 vs Ground truth scale normalized root:    -7.451607     9.879951
---------------------
Solving global hand scale which is
Solved global scale vs real global scale:     0.108607     0.090770
---------------------
   11     1
23 -0.0621625
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:     7.240659     6.911001
Root 2 vs Ground truth scale normalized root:    -8.195185     6.911001
---------------------
Solving global hand scale which is
Solved global scale vs real global scale:     0.110399     0.091926
---------------------
   12     0
24 -0.0702172
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:     6.068675     5.628127
Root 2 vs Ground truth scale normalized root:    -6.175155     5.628127
---------------------
Solving global hand scale which is
Solved global scale vs real global scale:     0.110438     0.093797
```

```
Solving global hand scale which is
Solved global scale vs real global scale:     0.111290     0.126024
---------------------
   19     1
   20     0
40 0.126782
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:     6.728361     7.524141
Root 2 vs Ground truth scale normalized root:    -4.892658     7.524141
---------------------
Solving global hand scale which is
Solved global scale vs real global scale:     0.110965     0.090761
---------------------
   20     1
   21     0
42 -0.0113637
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:     5.400281     5.359130
Root 2 vs Ground truth scale normalized root:    -5.337444     5.359130
---------------------
Solving global hand scale which is
Solved global scale vs real global scale:     0.104483     0.093709
---------------------
   21     1
43 0.0769343
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:     6.960660     7.445234
Root 2 vs Ground truth scale normalized root:    -2.686331     7.445234
---------------------
Solving global hand scale which is
Solved global scale vs real global scale:     0.104623     0.093724
---------------------
   22     0
44 -0.100865
---
Solving scale normalized zroot
Root 1 vs Ground truth scale normalized root:     9.155630     8.706736
Root 2 vs Ground truth scale normalized root:    -6.145841     8.706736
---------------------
Solving global hand scale which is
Solved global scale vs real global scale:     0.110954     0.125765
```

Final step

- Suppose that CNN $f(I) = \left\{\left(\widehat{u_k}, \widehat{v_k}, \widehat{Z_k^r}\right)\right\}_{k \in K}$ takes as input the cropped image I, and predicts the ***"normalized"*** scale normalized root-relative 2.5D pose (***"normalized"*** means normalized for CNN training: notation $\widetilde{\ }$)

- local scale-norm projection (ranges in [0, 1]) $\widetilde{\widehat{u_k}}$ $\widetilde{\widehat{v_k}}$
- normed root-relative scale-norm depth (ranges in, for example, [-1, 1]) $\widetilde{\widehat{Z_k^r}}$

Final step

- First off, take an unnormalization effort to convert CNN output to real scale-normalized root-relative 2.5D pose (for example, $\left(\widehat{u_k}, \widehat{v_k}, \widehat{Z_k^r}\right)$ = (480 pixel, 220 pixel, 14cm))

$$g(I) = \left\{\left(\widehat{u_k}, \widehat{v_k}, \widehat{Z_k^r}\right)\right\}_{k\in K} = unnorm\left(f(I)\right) = unnorm(\widetilde{\left\{\left(\widehat{u_k}, \widehat{v_k}, \widehat{Z_k^r}\right)\right\}_{k\in K}})$$

- Afterwards, compute scale-normalized root depth $\widehat{Z_{root}}$, add each root-relative depth prediction $\widehat{Z_k^r}$ with $\widehat{Z_{root}}$

$$\widehat{Z_k} = \widehat{Z_k^r} + \widehat{Z_{root}}$$

Final step

- Get scale normalized absolute $\widehat{X_k}, \widehat{Y_k}$

$$\widehat{X_k} = \frac{(\widehat{u_k} - u_0)\widehat{Z_k^r}}{f_x}, \widehat{Y_k} = \frac{(\widehat{v_k} - v_0)\widehat{Z_k^r}}{f_y}$$

As you can see, $\widehat{X_k}$ and $\widehat{Y_k}$ are computed using 2.5D $\widehat{u_k}$ and $\widehat{v_k}$ and $\widehat{Z_k^r}$

$\widehat{Z_k}$ is computed using $\widehat{Z_k^r}$ and previously got $\widehat{Z_{root}}$

- To this end, we have acquired scale-normalized absolute 3D joint in camera frame (not root-relative any more)

$$\hat{P} = \left\{\widehat{P_k}\right\}_{k \in K} = \left\{(\widehat{X_k}, \widehat{Y_k}, \widehat{Z_k})\right\}_{k \in K}$$

Final step

- Use the statistics on average bone length (each bone) $\{\mu_{kl}\}_{k,l\in\varepsilon}$,

And scale-normalized absolute 3D pose $\hat{P}$

We solve the hand scale "$scale$"

- Finally, we come to the solution

$$P = \{P_k\}_{k\in K} = \left\{\frac{scale}{C}\widehat{P_k}\right\}_{k\in K}$$

- This is the global 3D in camera frame,
  $camera\ matrix\ \times global\ 3D = global\ projection\ in\ raw\ iamge$