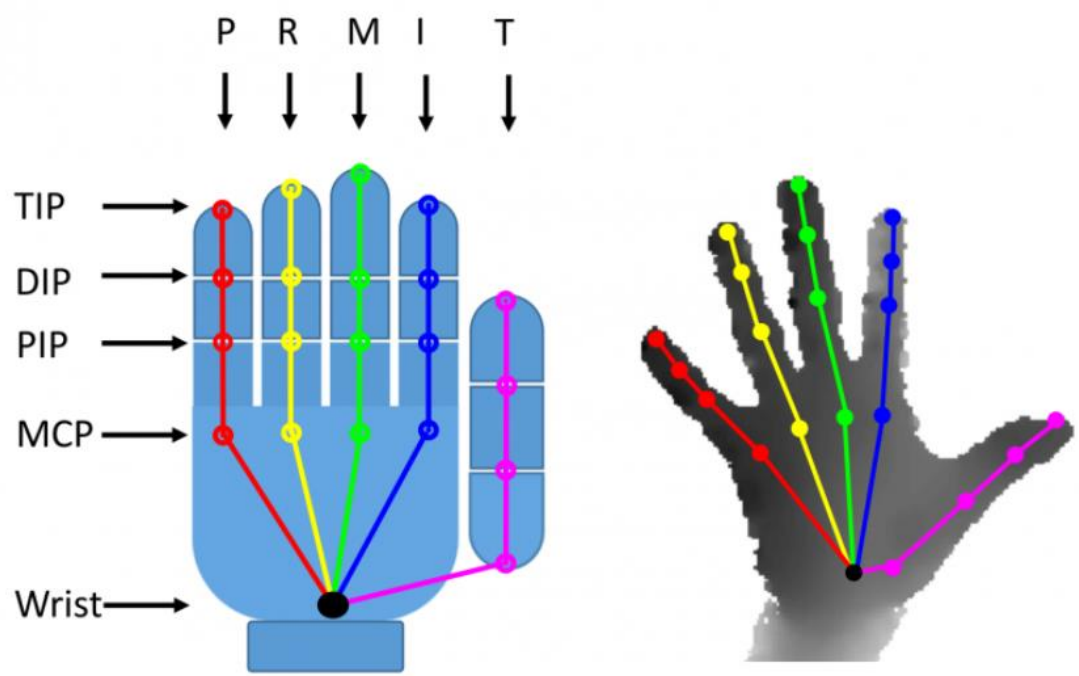
V2V-PoseNet: Concept and Modifications for HANDS 2019 Challenge

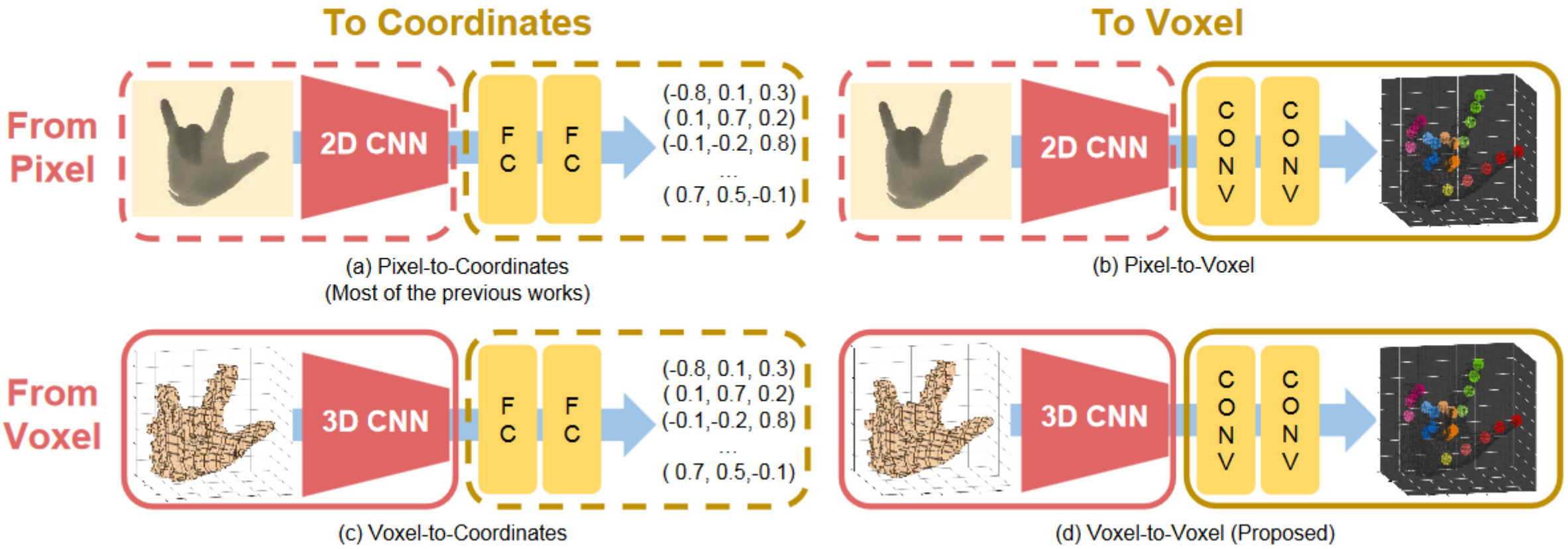
Marek Hrúz

Hand Pose Estimation

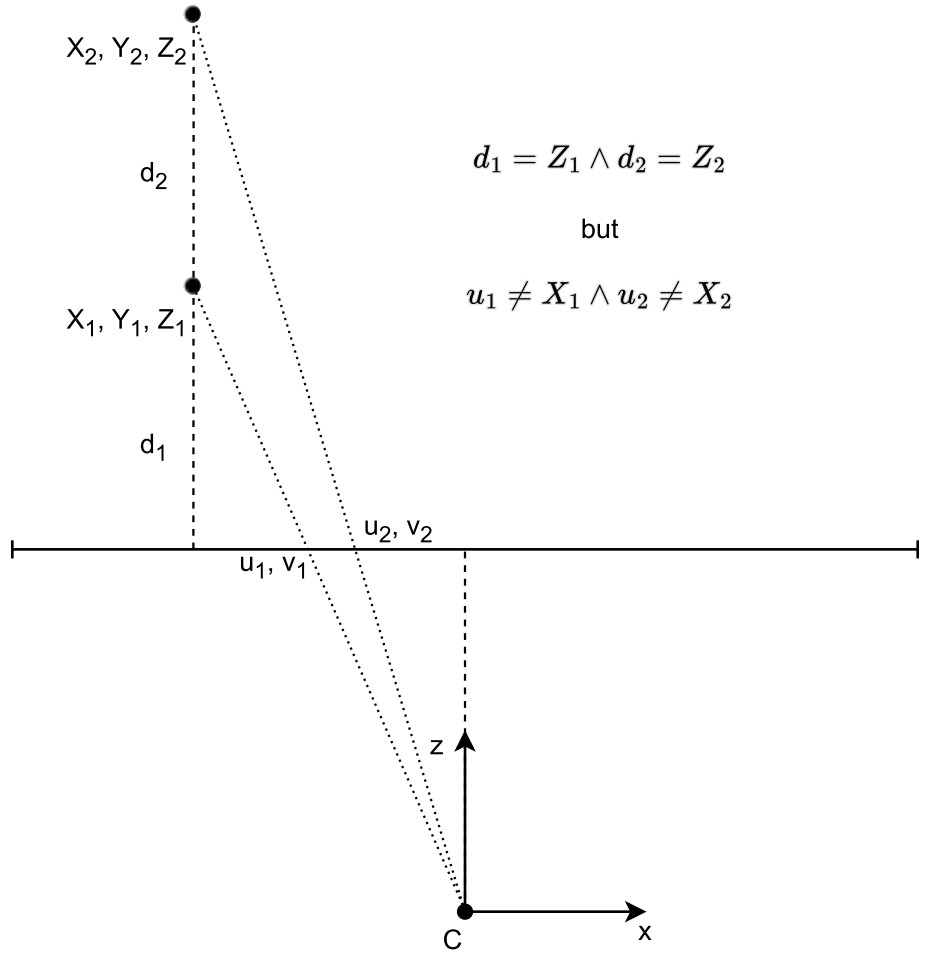
- The task of finding a 3D position of individual joints
- Used in VR/AR application
- Gesture Recognition
- Sign Language Recognition
- Contactless control



Input/Output Representation



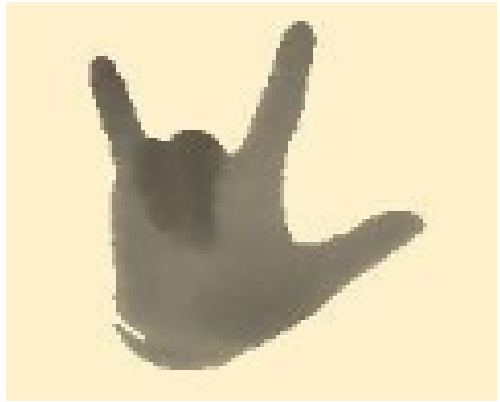
3D vs 2.5D projection

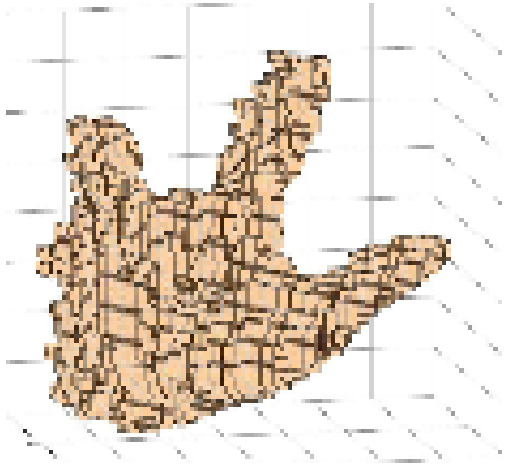


$$d_1 = Z_1 \wedge d_2 = Z_2$$

but

$$u_1 \neq X_1 \wedge u_2 \neq X_2$$





Volumetric representation of the input

- Each pixel of the depth map is reprojected into 3D (using known projective parameters)
- The resulting 3D space is discretised into a grid
- A predetermined cube is drawn around the refined point
- The volumetric representation has 1 when a depth point lies inside the given grid field (at least a little) and 0 otherwise

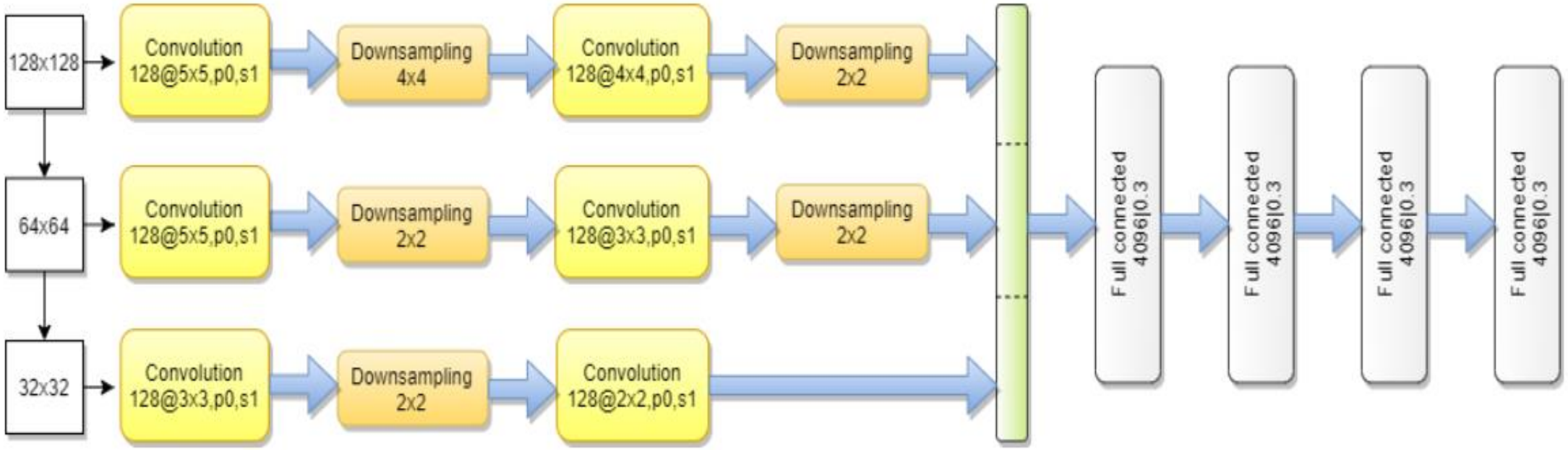
Volumetric representation of the output

- The output is a 3D heat-map of the ground truth position of joints
- For each joint there is one 3D heat-map
- Heat map is a Gaussian centred on the GT position of the joint
- Formally:

$$H_{\mathrm{n}}^{*}(i,j,k) = \exp\left(-\frac{(i-i_{\mathrm{n}})^2 + (j-j_{\mathrm{n}})^2 + (k-k_{\mathrm{n}})^2}{2\sigma^2}\right), \tag{1}$$

where H_n^* is the ground-truth 3D heatmap of nth keypoint, (i_n,j_n,k_n) is the ground-truth voxel coordinate of nth keypoint, and $\sigma = 1.7$ is the standard deviation of the Gaussian peak.

Reference point refinement



Network architecture

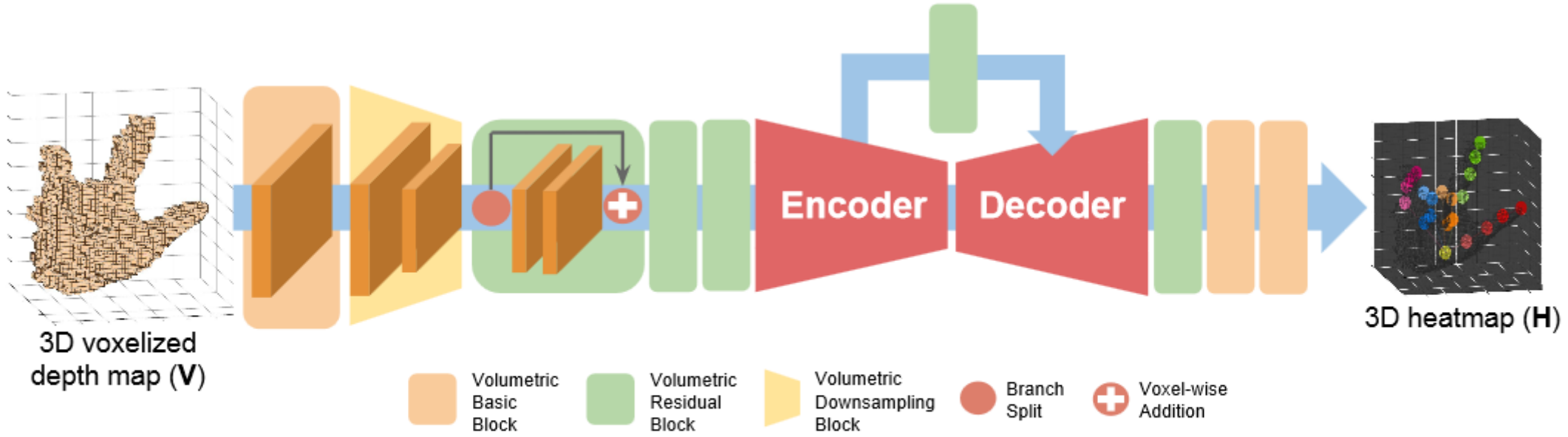
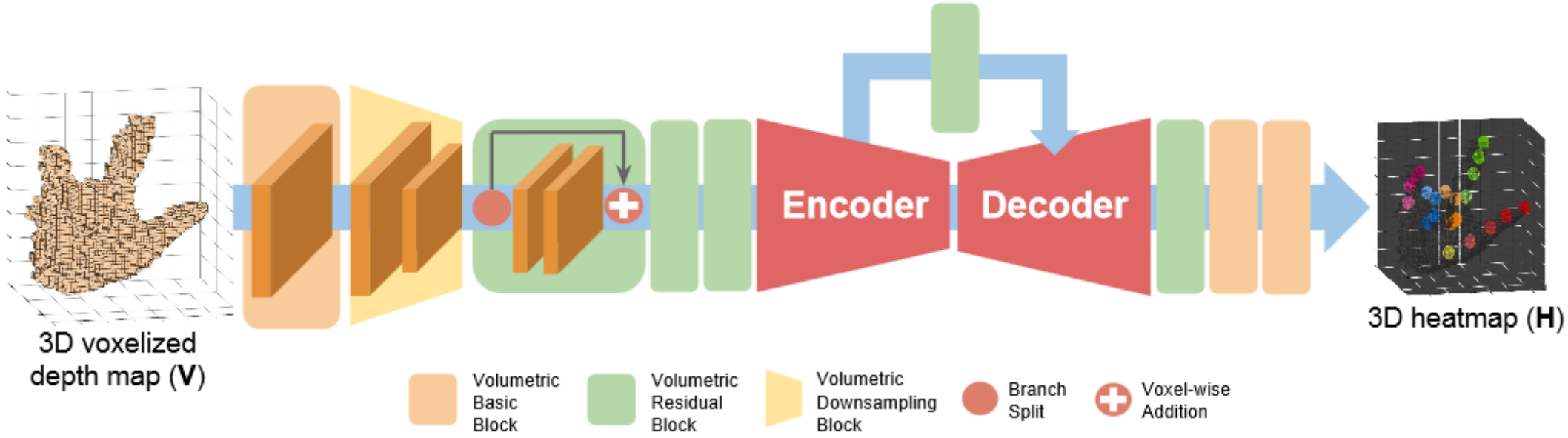


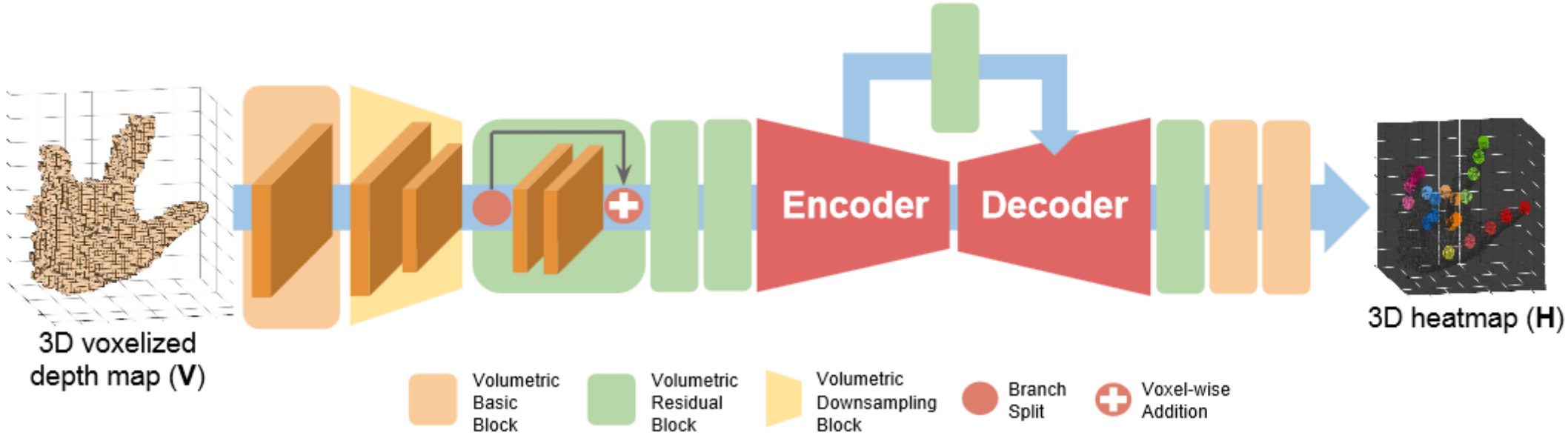
Figure 3: Overall architecture of the V2V-PoseNet. V2V-PoseNet takes voxelized input and estimates the per-voxel likelihood for each keypoint through encoder and decoder. To simplify the figure, we plotted each feature map without Z-axis and combined the 3D heatmaps of all keypoints in a single volume. Each color in the 3D heatmap indicates keypoints in the same finger.

Volumetric basic block

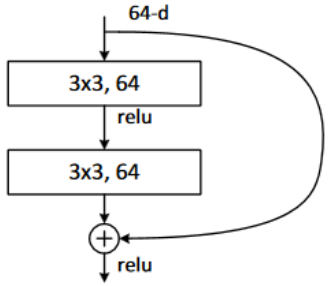


- 3D Conv – (16@7x7x7, 32@1x1x1, 32@1x1x1)
- Volumetric Batch Normalization (normal one, no special nothing)
- ReLU activation (after normalization!)

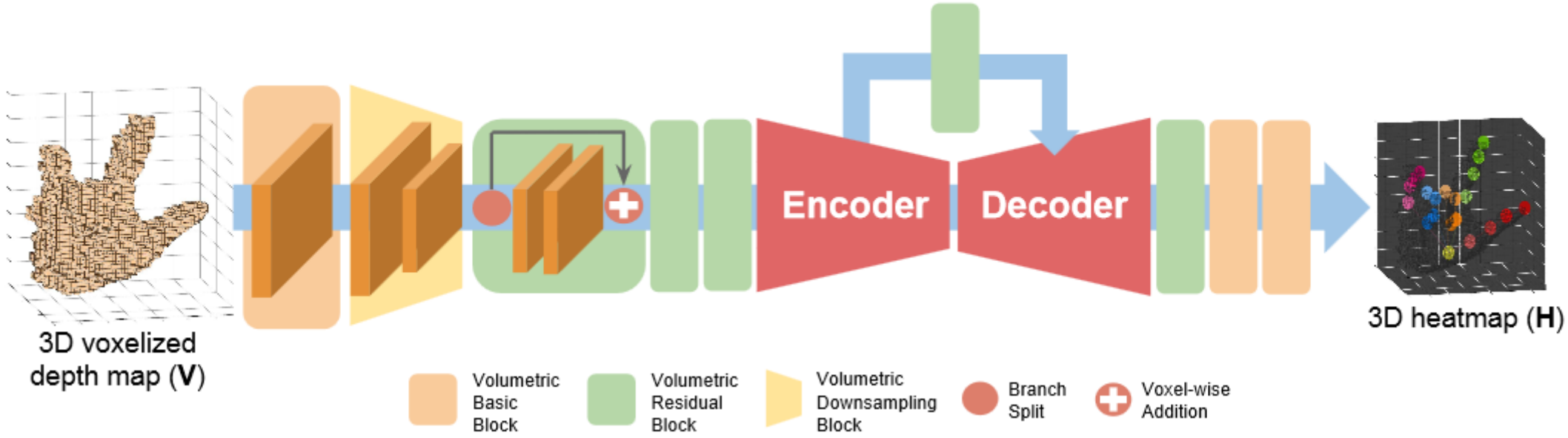
Volumetric residual block



- One of ResNet options (projection shortcuts opt B)
- Conv3D (3x3x3) → VBN → ReLU → Conv3D (3x3x3) → VBN → concat
- Concat:
  - if inDim == outDim → identity
  - else → Conv3D (1x1x1)

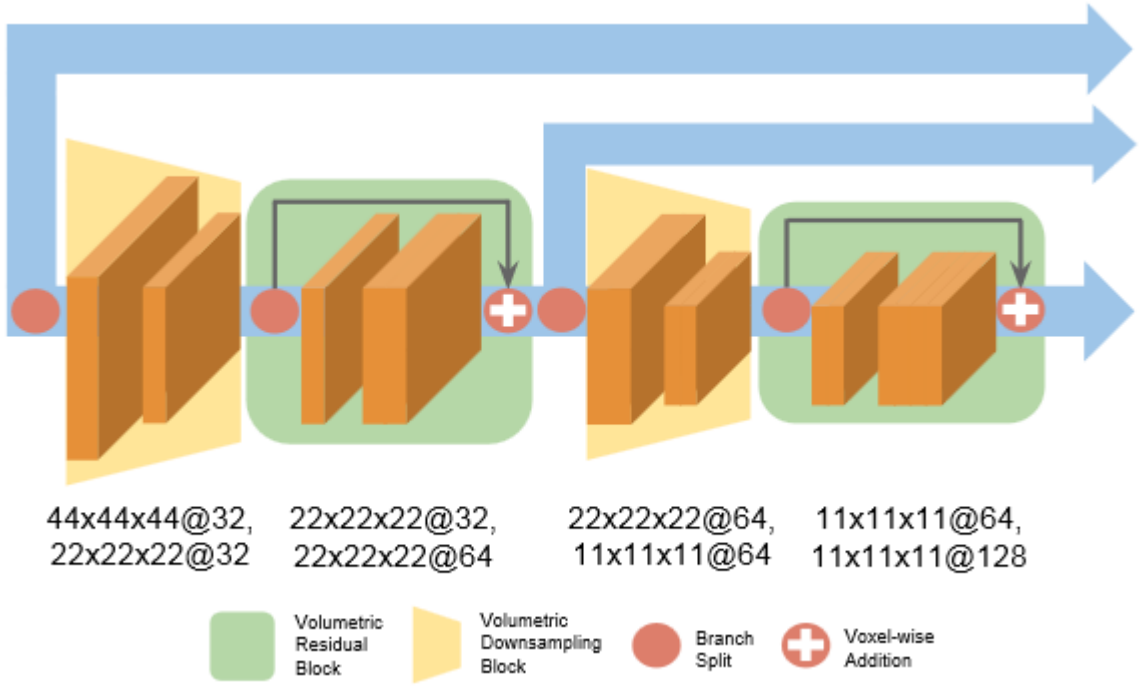


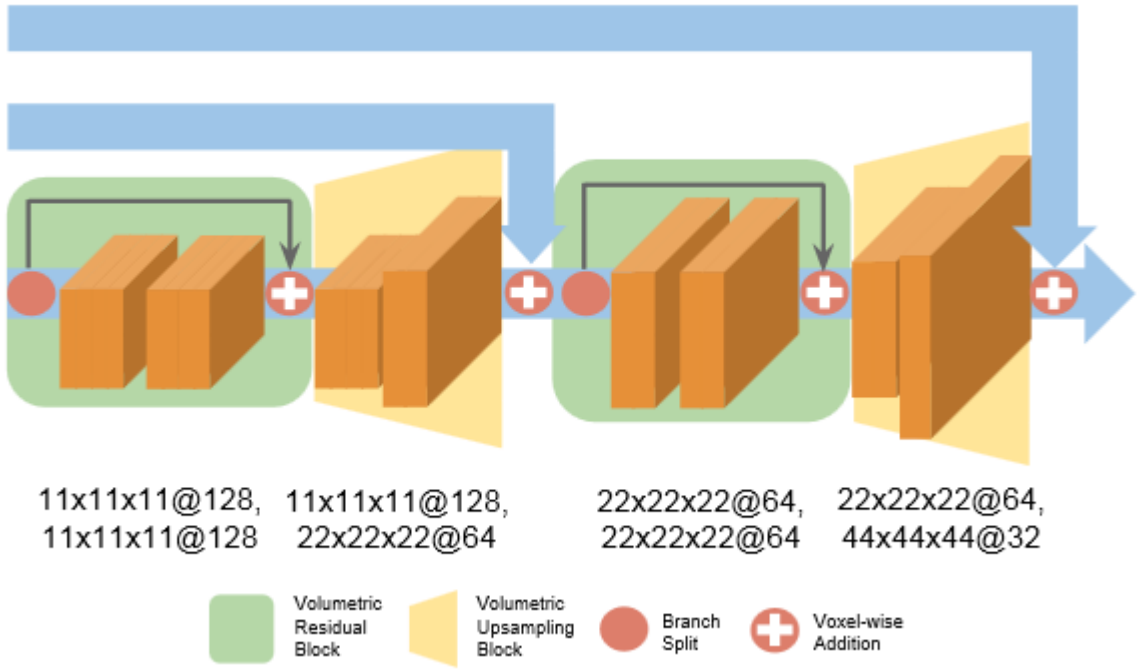
Volumetric down/up sampling block



- Downsampling is 3D max pool (2x2x2, stride 2x2x2)
- Upsampling is 3D deconv + VBN

Encoder/Decoder





Learning

- MSE – elementwise, predicted volume vs GT volume
- Gaussian init (mean = 0, std = 0.001)
- RMSProp, batch size 8, lr = 0.00025
- Input size = 88 x 88 x 88
- Augmentation – rotation XY, 3D scaling, 3D translation
- Torch7 – 10 epochs

Results

Input \Output	3D Coordinates	Per-voxel likelihood
2D depth map	18.85 (21.1 M)	13.01 (4.6 M)
3D voxelized grid	16.78 (457.5 M)	**10.37 (3.4 M)**

Table 1: Average 3D distance error (mm) and number of parameter comparison of the input and output types in the NYU dataset. The number in the parenthesis denotes the number of parameters. The visualized model for each input and output type is shown in Figure 2.

Methods	Mean error (mm)
LRF	12.58
DeepModel	11.56
Hand3D	10.9
CDO	10.5
DeepPrior	10.4
CrossingNets	10.2
Cascade	9.9
JTSC	9.16
DeepPrior++	8.1
REN-4x6x6	7.63
REN-9x6x6	7.31
Pose-REN	6.79
V2V-PoseNet (Ours)	**6.28**

(a) ICVL

Methods	Mean error (mm)
DISCO	20.7
DeepPrior	19.73
Hand3D	17.6
DeepModel	17.04
JTSC	16.8
Feedback	15.97
Global-to-Local	15.60
Lie-X	14.51
3DCNN	14.1
REN-4x6x6	13.39
REN-9x6x6	12.69
DeepPrior++	12.24
Pose-REN	11.81
V2V-PoseNet (Ours)	**8.42**

(b) NYU

Methods	Mean error (mm)
Cascade	15.2
Cls-Guide	13.7
MultiView	13.2
Occlusion	12.8
CrossingNets	12.2
REN-9x6x6	9.7
DeepPrior++	9.5
Pose-REN	8.65
V2V-PoseNet (Ours)	**7.49**

(c) MSRA

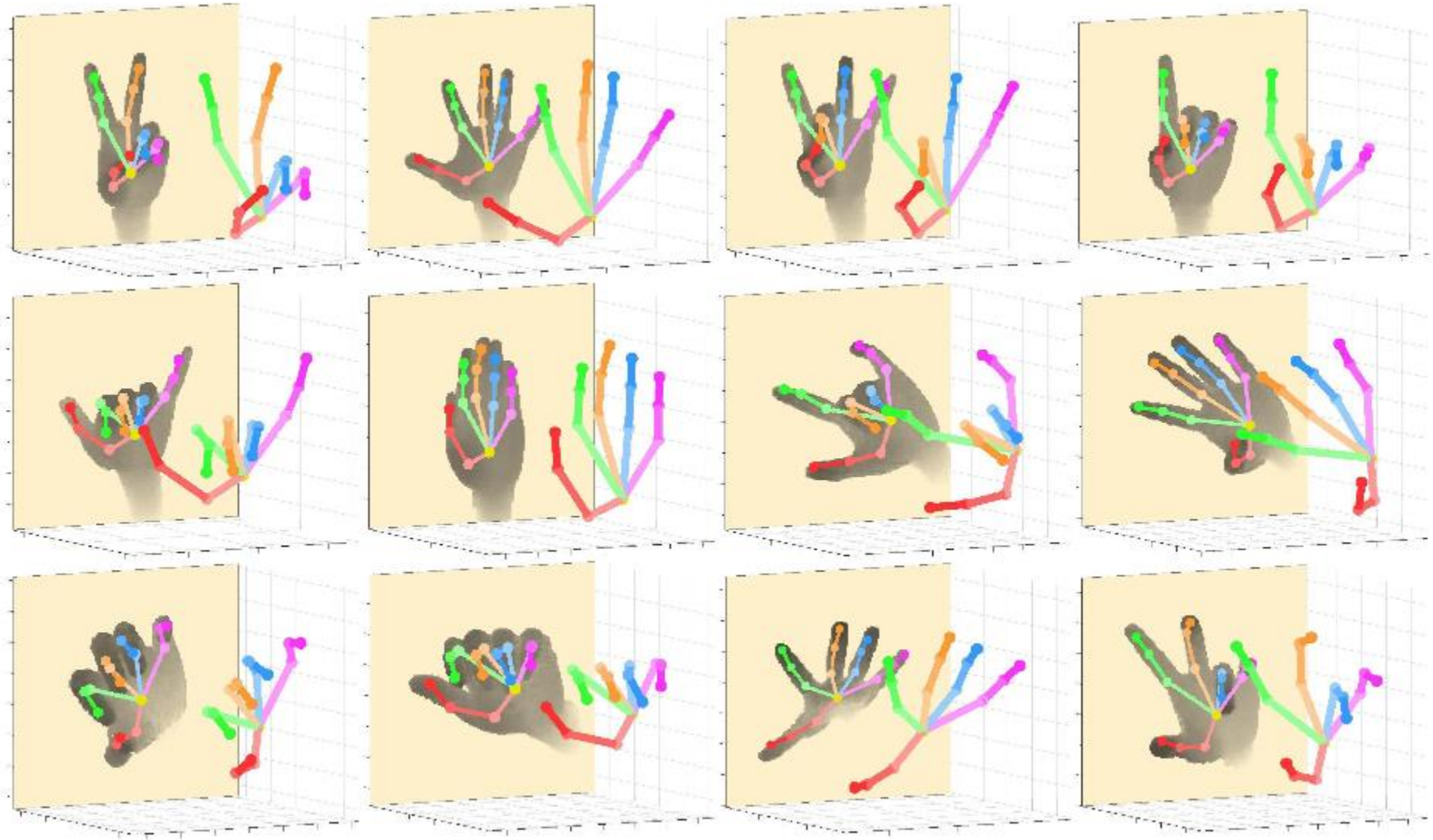


Figure 9: Qualitative results of our V2V-PoseNet on the ICVL dataset. Backgrounds are removed to make them visually pleasing.

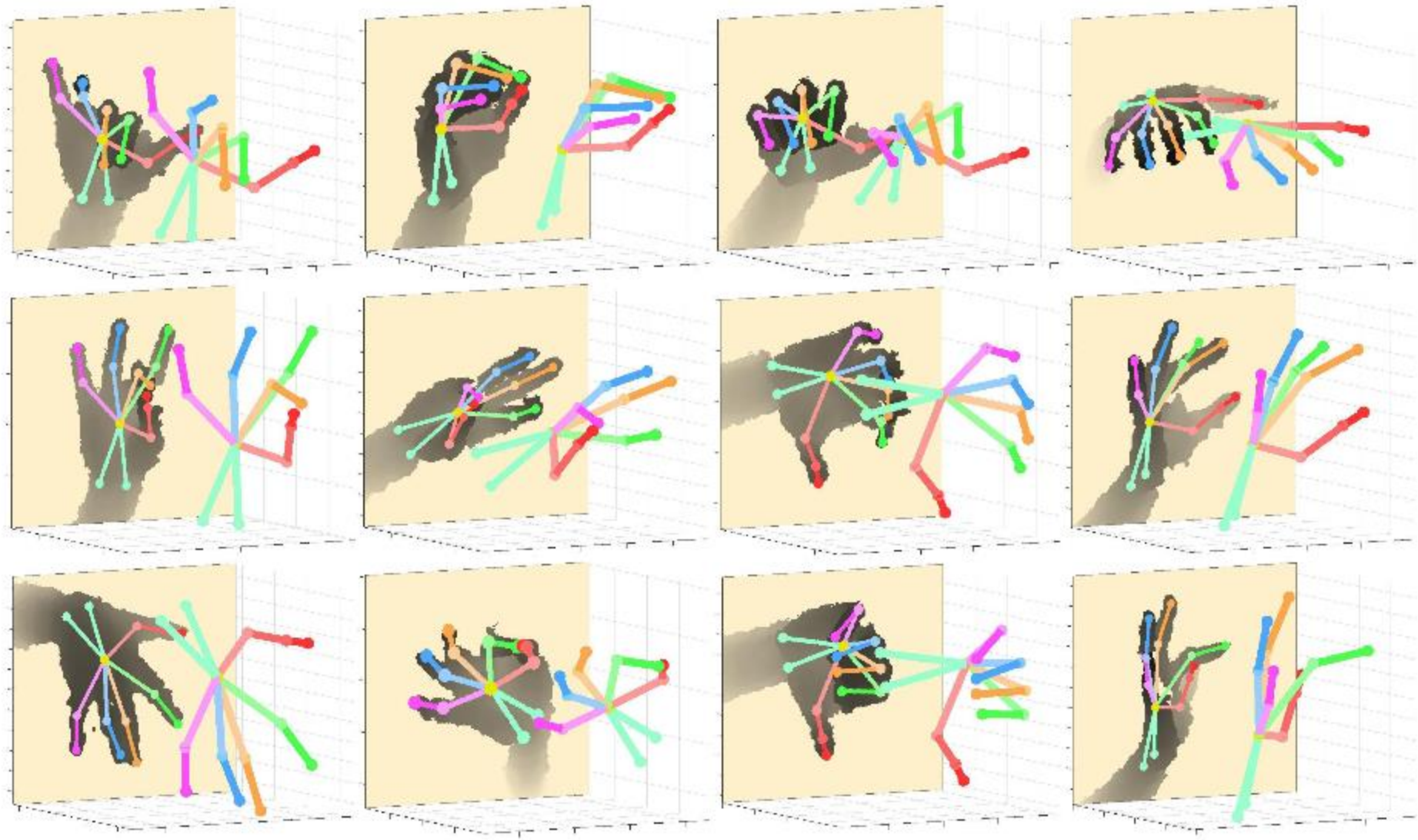


Figure 10: Qualitative results of our V2V-PoseNet on the NYU dataset. Backgrounds are removed to make them visually pleasing.

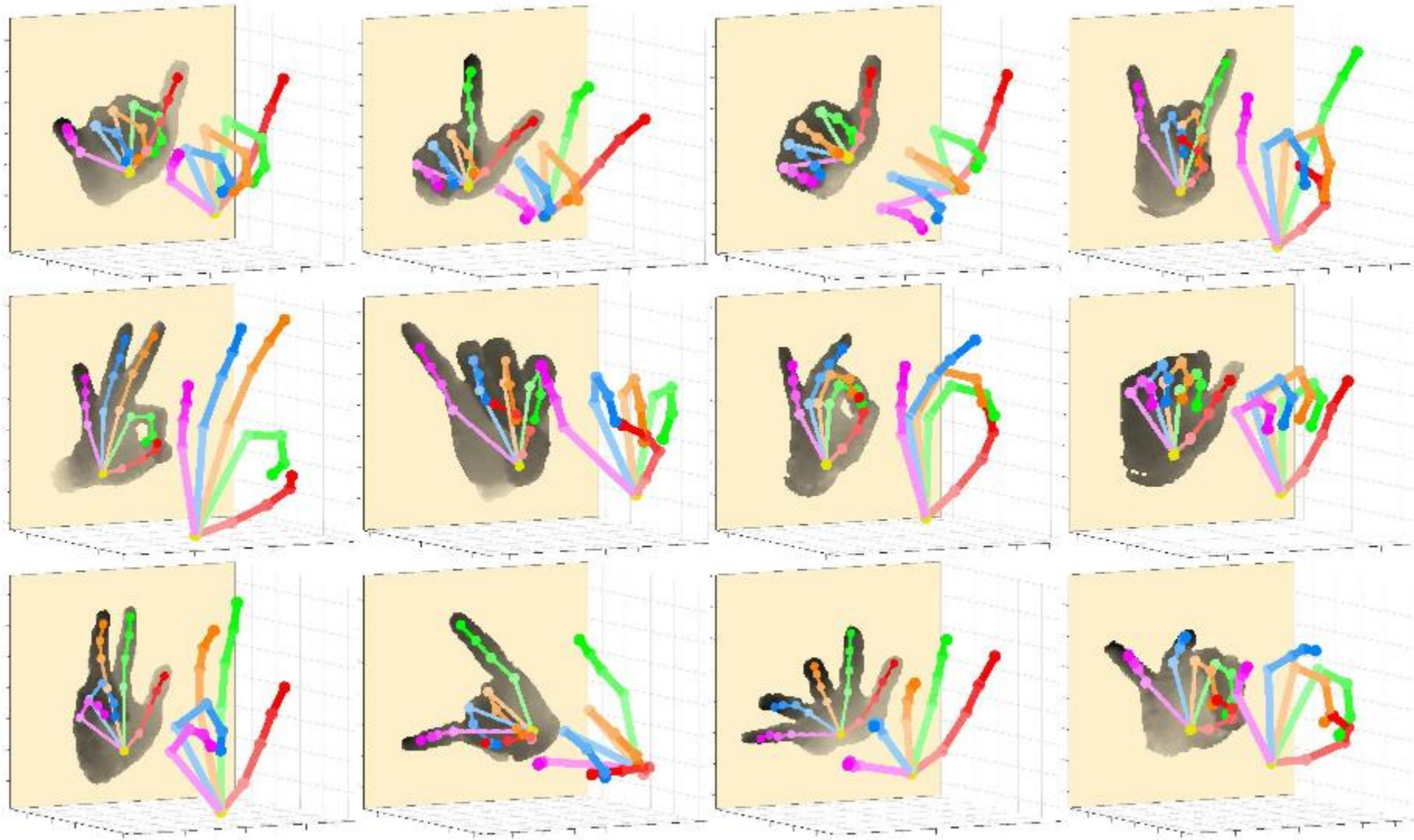


Figure 11: Qualitative results of our V2V-PoseNet on the MSRA dataset. Backgrounds are removed to make them visually pleasing.

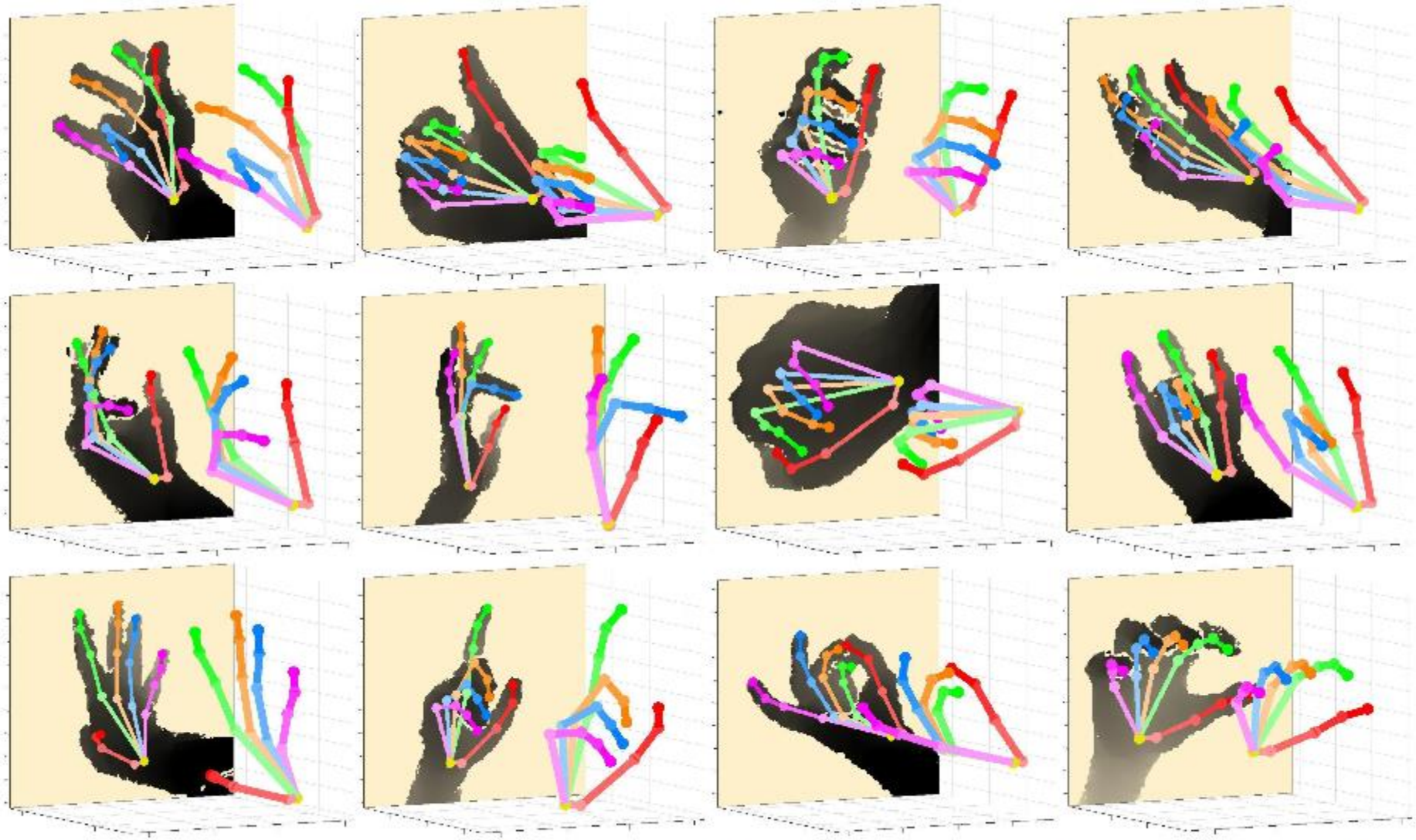
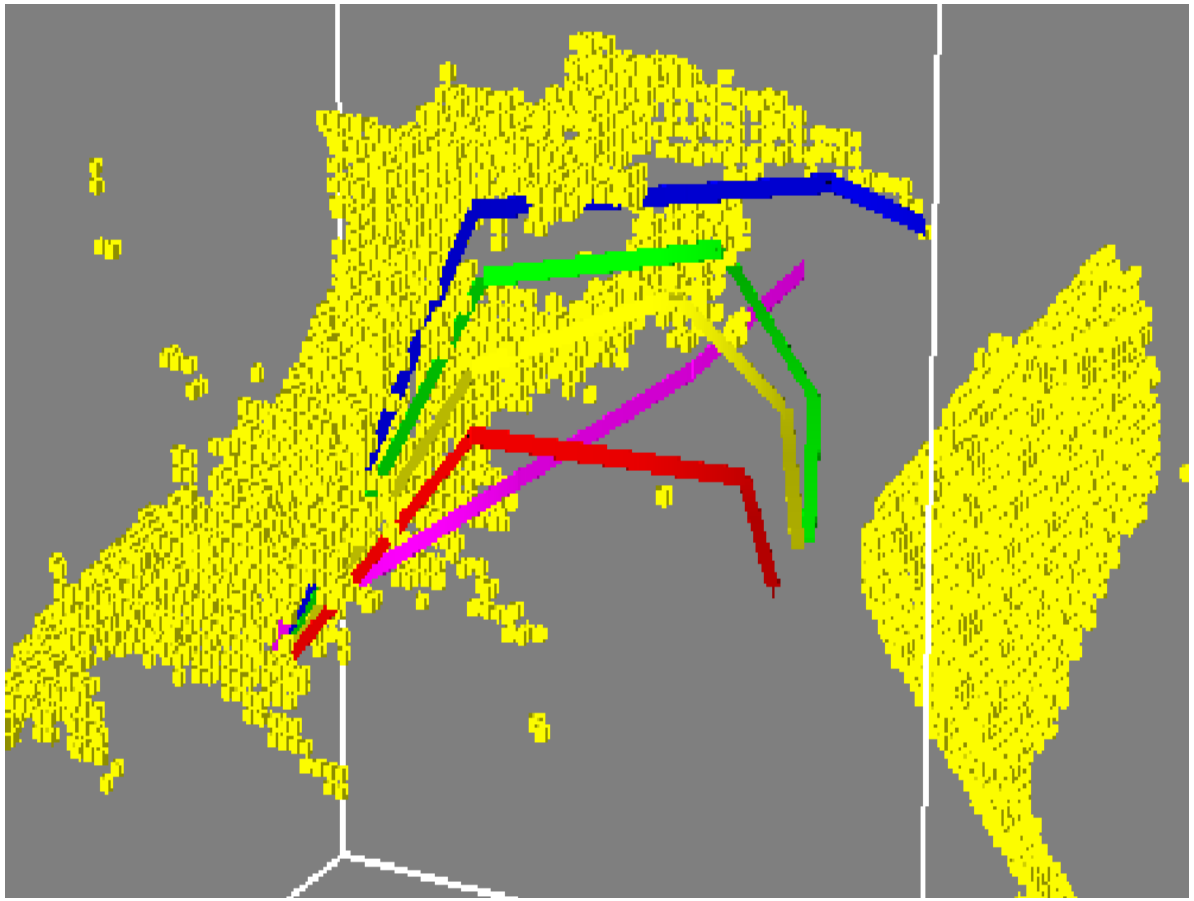
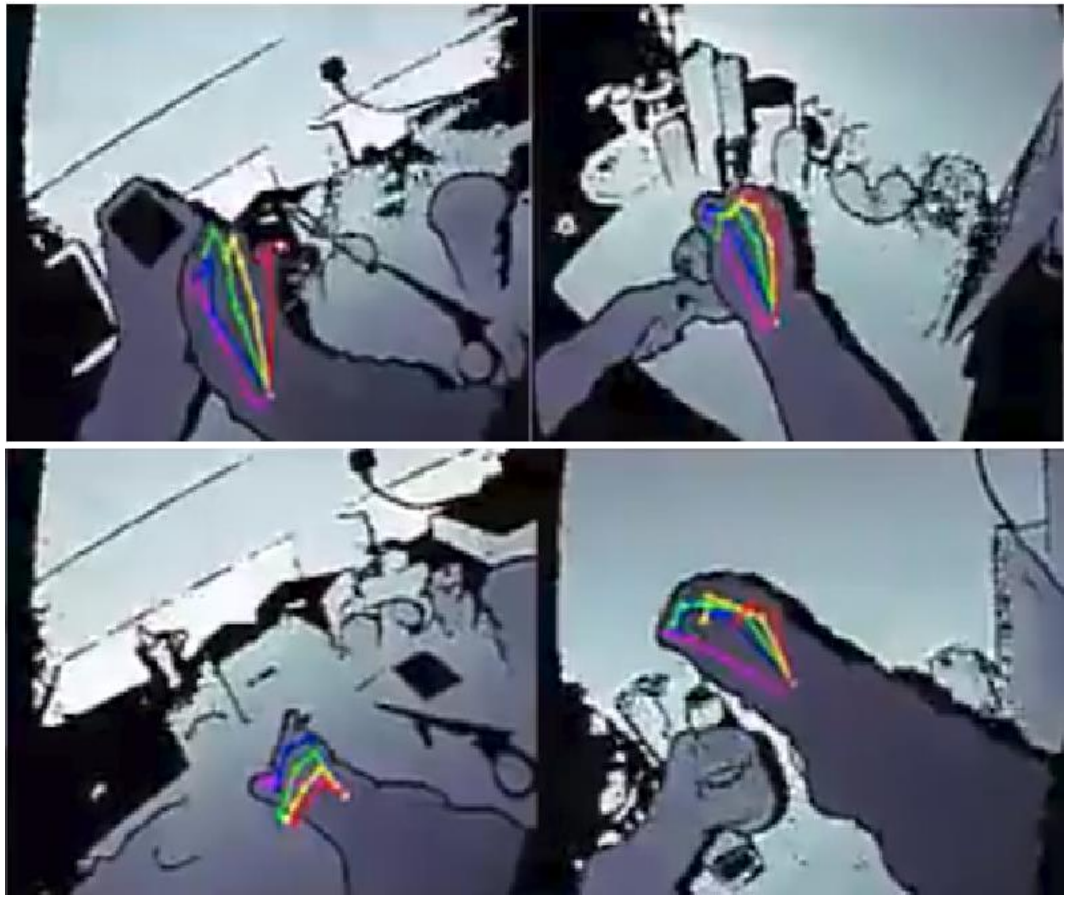


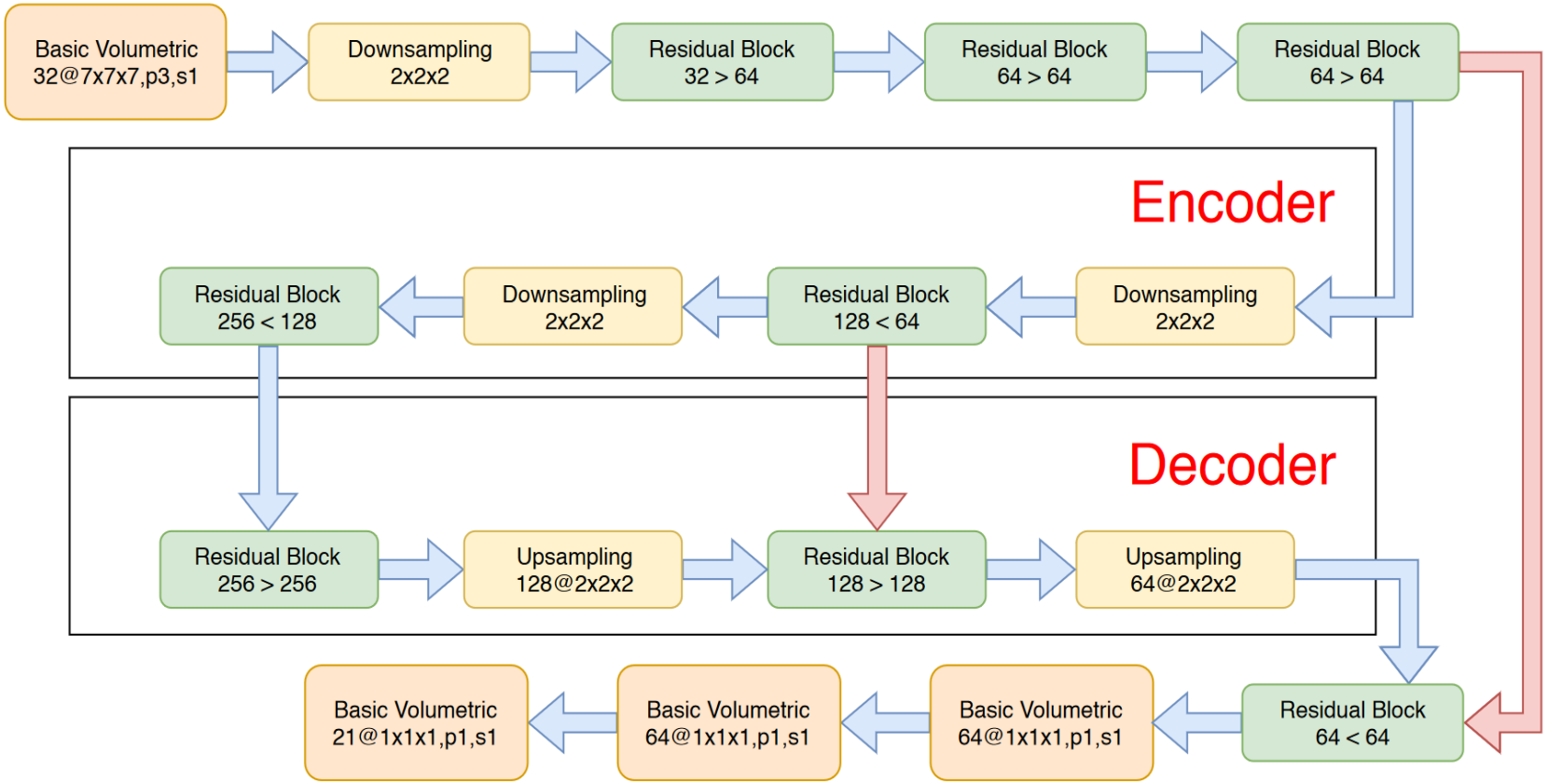
Figure 12: Qualitative results of our V2V-PoseNet on the HANDS 2017 frame-based 3D hand pose estimation challenge dataset. Backgrounds are removed to make them visually pleasing.

Hands 2019 Task 2 – Egocentric view of Hands interacting with objects





Our modified architecture



Width n	Epoch	Precision [mm]
16	10	40.04
32	10	38.18
64	10	38.05
84	10	71.99
16	20	39.61
32	20	38.03
64	**20**	**37.11**
84	20	70.34

Post-processing

- Hand Joint Location Estimation
  - In each output channel we find the global maximum
  - We use a sub-voxel precision computed as weighted average around this maximum
- Prediction Ensemble
  - We combine up to 100 best location estimation and 5 epochs
- Pose prior
  - Modeled by Truncated SVD computed on GT data
- Temporal context
  - Averaging coordinates

Refinement	Epoch	Max	S-max	Gauss
RoM	10	42.78	42.70	41.19
CoM	10	40.13	40.04	38.97
RoM	20	42.05	41.97	39.86
CoM	20	39.69	39.61	**38.08**

Epoch Ensemble	# N-best	Precision [mm]
20	-	37.17
20	100	36.14
1-20	100	35.18
3-best	100	34.88
5-best	100	**34.85**
7-best	100	34.90

Epoch Ensemble	N-best	Window Size	Precision [mm]
5-best	100	+1, -1	33.23
5-best	100	+3, -3	**33.11**
5-best	100	+5, -5	33.12
5-best	100	+7, -7	33.16
5-best	100	+10, -10	33.19

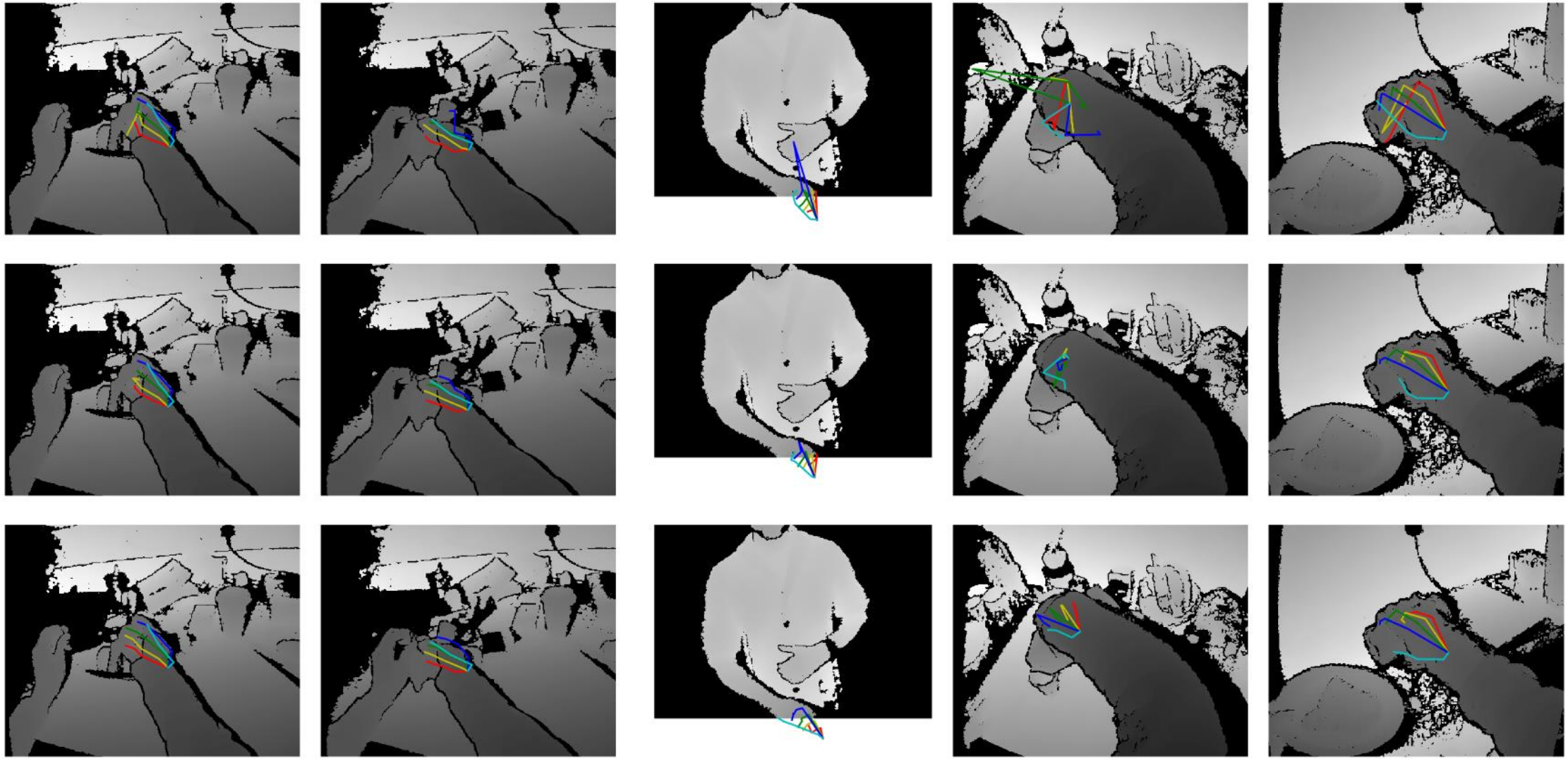


FIGURE 7: Example of the testing data prediction. The first row represents the raw output of the pose network obtained by the S-max approach (best epoch on the test set). The second row represents the effect of the ensemble (5 best epochs, N-best locations). The third row represents the ensemble + post-processing effect (TruncatedSVD pose prior, temporal context).

Thanks for your attention!