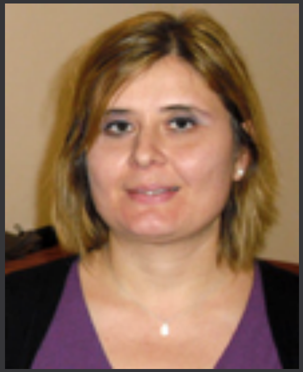
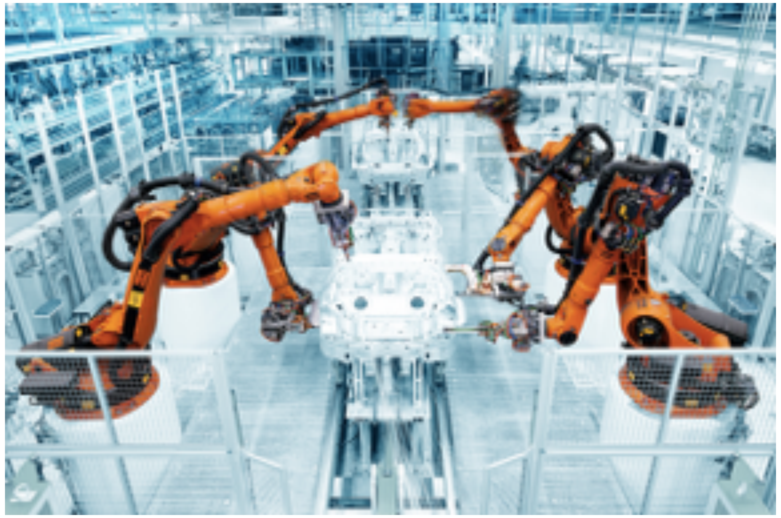
Efficient Real-Time Pixelwise Object Class Labeling for Safe Human-Robot Collaboration in Industrial Domain






Vivek Sharma, Frank Dittrich, Sule Yildirim-Yayilgan, and Luc Van Gool

Problem Statement



Domain: Scene Analysis for Safe-Human-Robot-Collaboration
This work builds on top of our previous work (Sharma *et al.,* 2015) and (Dittrich *et al.*, 2014).

Schematic Layout

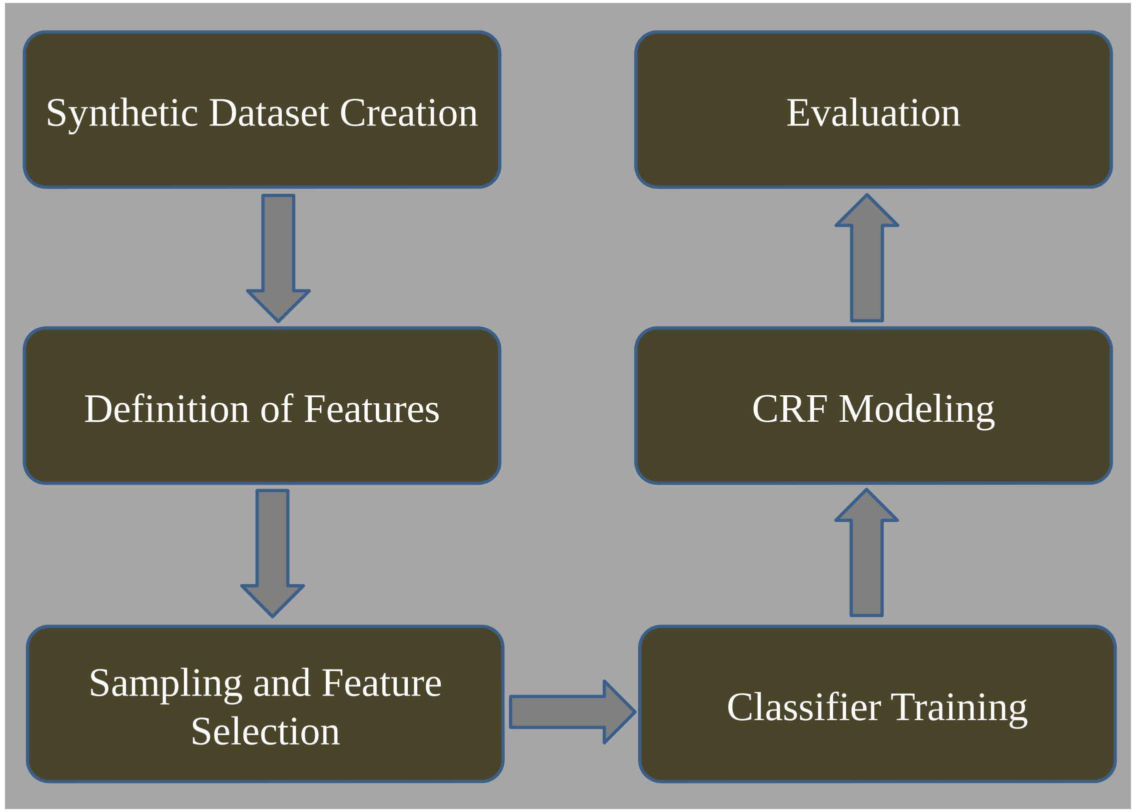


Figure 1: Schematic layout of the pixelwise object class segmentation system.

Collection of Data

- **Synthetic Data Generated**:
  - **Depth** Image with additive white Gaussian noise.
  - **RGB** Image (groundtruth).
  - **Data Instances**: Background, human(head, body, upper-arm, lower-arm, hands, legs), chair, plant ,same class (**table and storage**)
  - **Unlimited amount of data can be generated.**
    - 640X480{1(Depth,Float),3(RGB,Integer)}

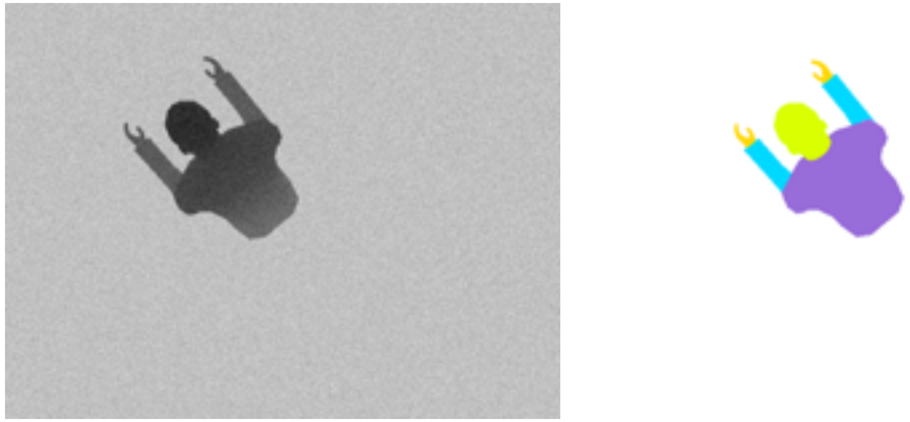


Figure 2: Synthetic generated depth data and it‘s corresponding ground truth image.

Robot Simulator

- V-REP
    - Virtual Robot Experimentation Platform (Fresse *et al.*, 2010)
        - Integrated Development Environment (IDE)
        - Distributed Control Architecture
        - Remote API Client
        - Supports: C/C++, Python, Lua, Java, Matlab, Octave or Urbi
        - Free for academic and research purposes

Human Multicolor Data

- Real world choreographies via KINECT skeleton tracking data from a calibrated multi-sensor setup.
- Synthetic representation of 3D human model based on a set of spheres in virtual environment (V-REP)
- Scaling factor for height ranging between 160-190 cm‘s.

$$S_{scaled} = \lambda \ x \ S_{original}$$

$$\{\lambda_{min} \text{x} 168=160,\ \lambda_{max} \text{x} 168=190\}$$

- For testing data ground truth, we use Automatic Annotation approach.

Setup

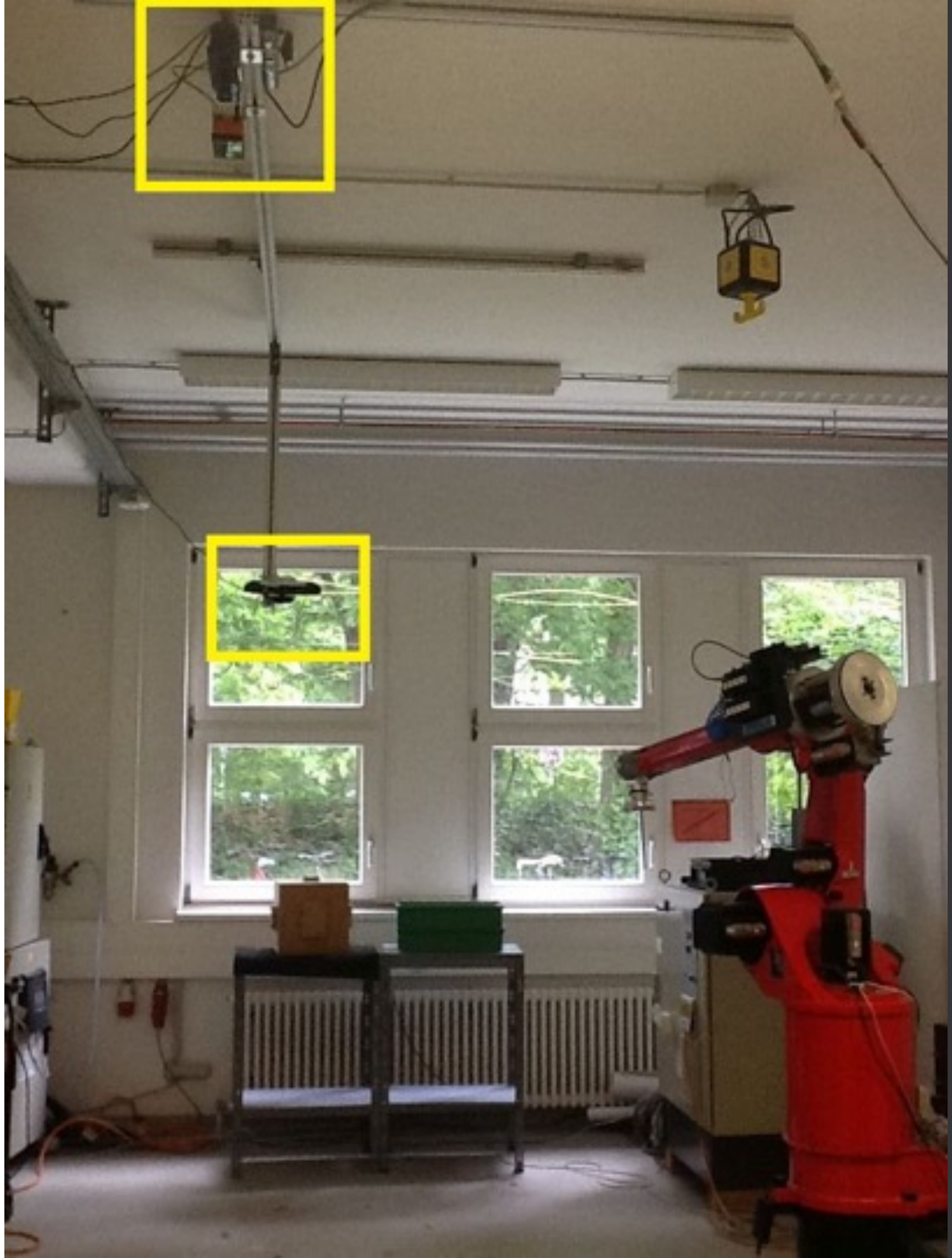


Figure 3: KINECT skeleton tracking setup.

Training Data: Human

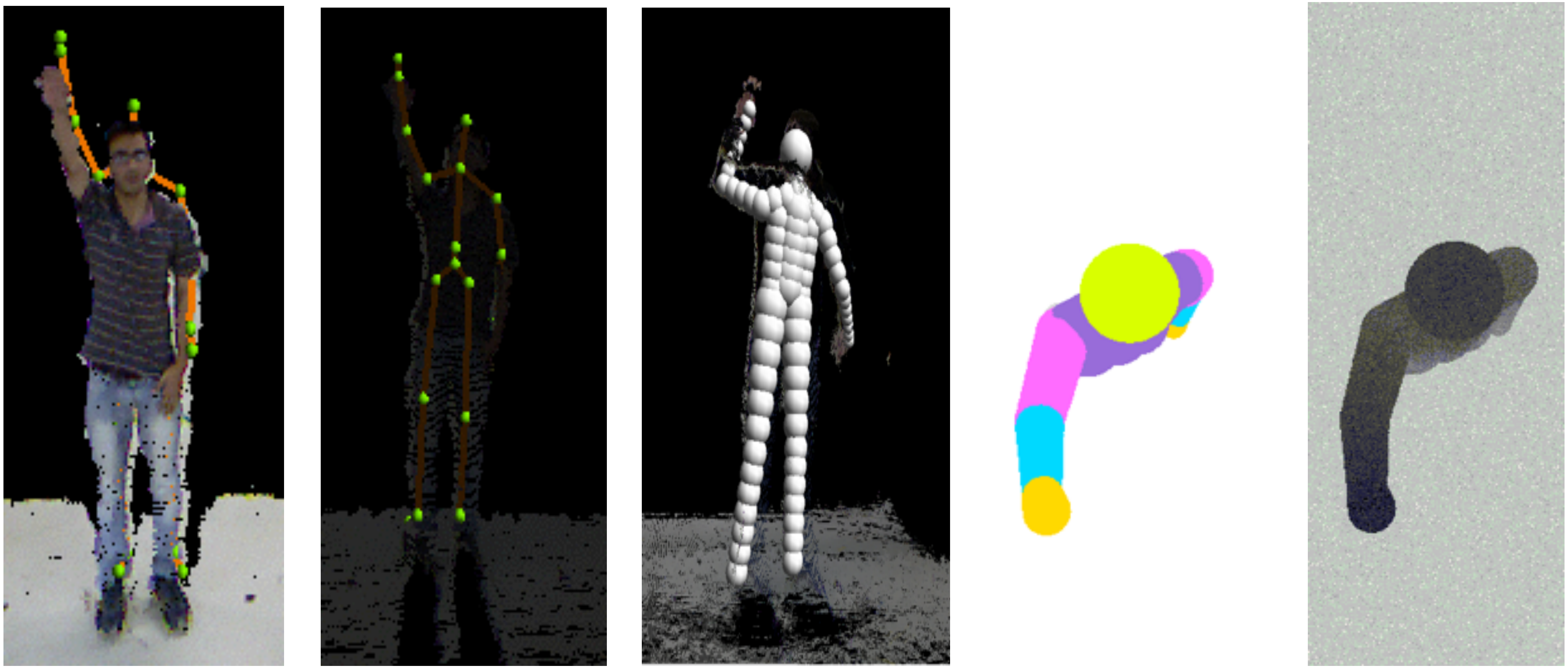


Figure 4: *Left:* KINECT skeleton tracking. *Center:* Coarse approximation of the human body, modeled by small set of spheres arraged along the skeleton estimate. *Right:* Finer sphere approximation of the human body, modeled on the spheres in the V-REP environment.

Training Data: Human

Figure 5: Synthetic depth data generated with a snythetic KINECT sensor of human, groundtruth(*left*) and synthetic depth frame with additive white Gaussian Noise(*right*).

Training Data: Chairs

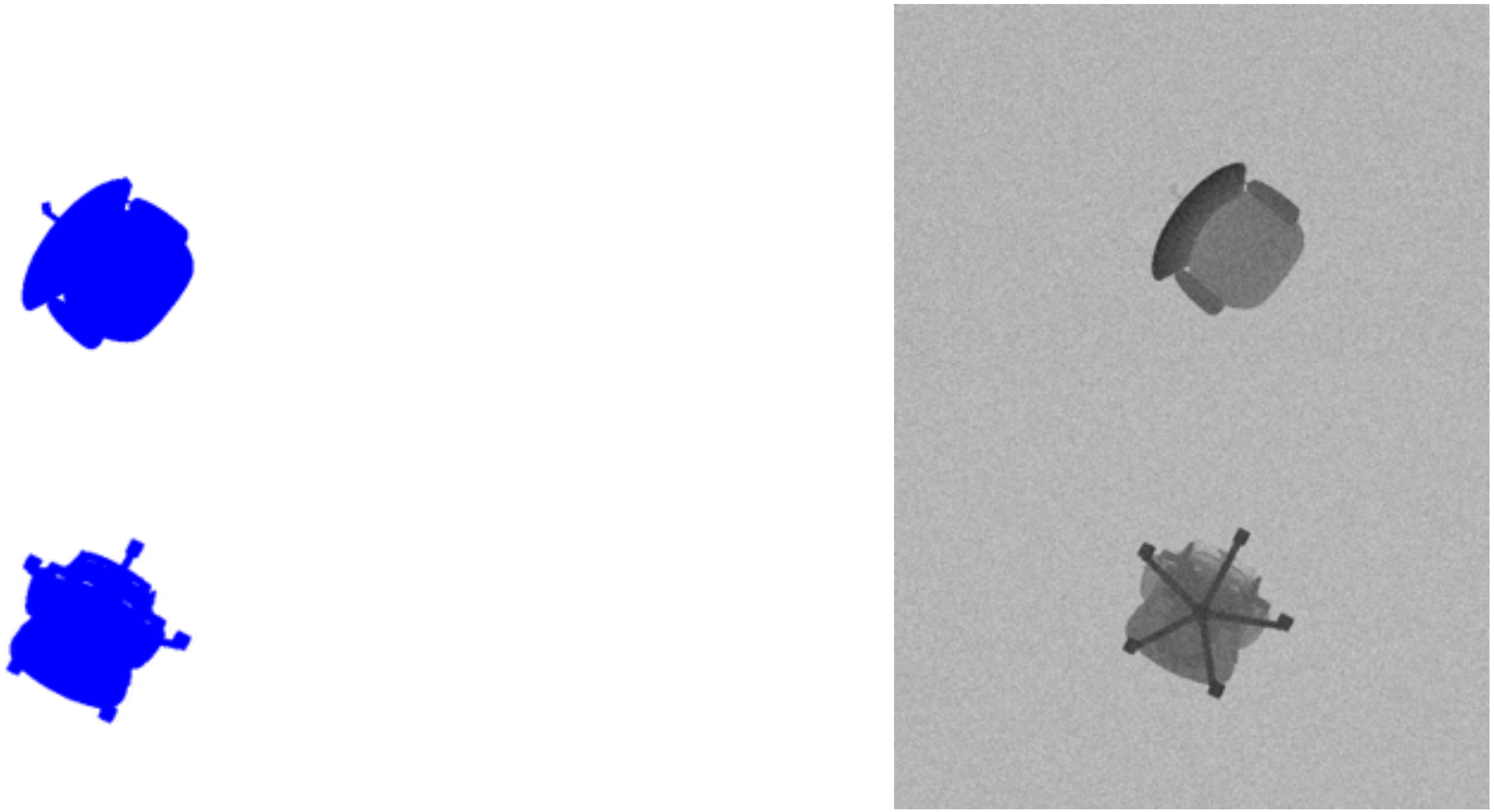


Figure 6: Synthetic depth data generated with a snythetic KINECT sensor of chairs, groundtruth(*left*) and synthetic depth frame with additive white Gaussian Noise(*right*).

Training Data : Tables

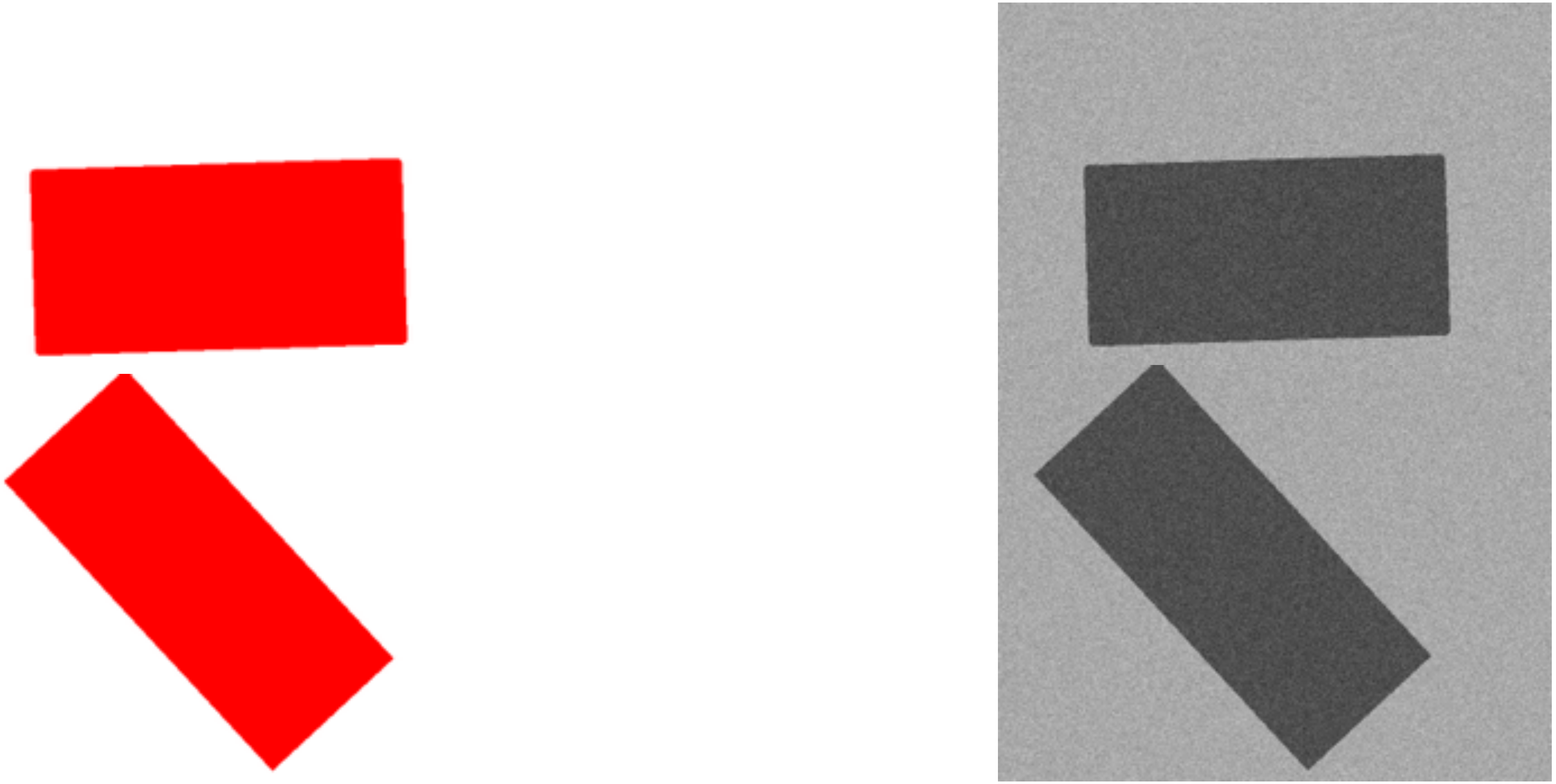


Figure 7: Synthetic depth data generated with a snythetic KINECT sensor of tables, groundtruth(*left*) and synthetic depth frame with additive white Gaussian Noise(*right*).

Training Data: Storages

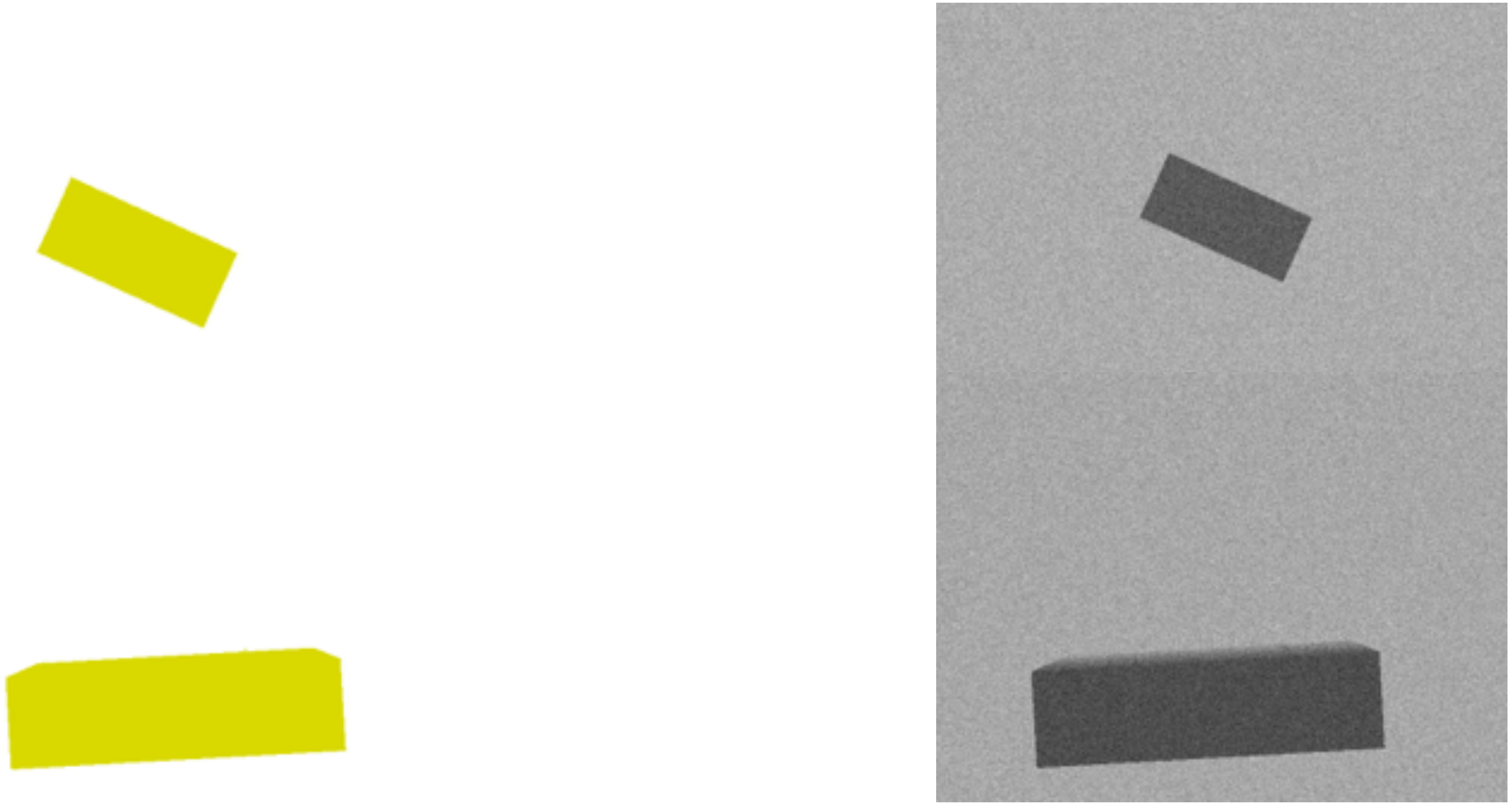


Figure 8: Synthetic depth data generated with a snythetic KINECT sensor of storages, groundtruth(*left*) and synthetic depth frame with additive white Gaussian Noise(*right*).

Testing Data: Plants

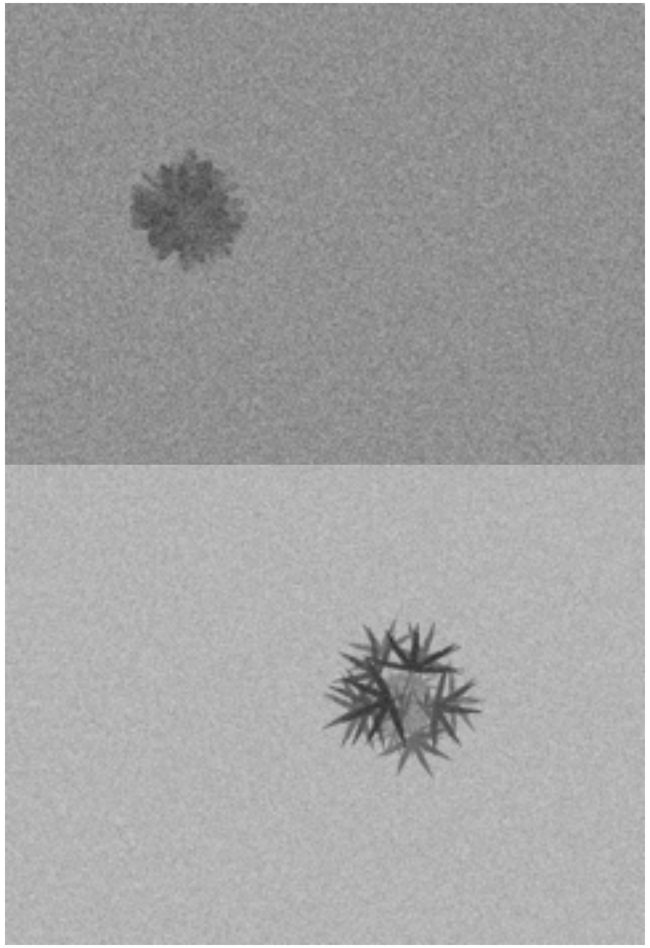
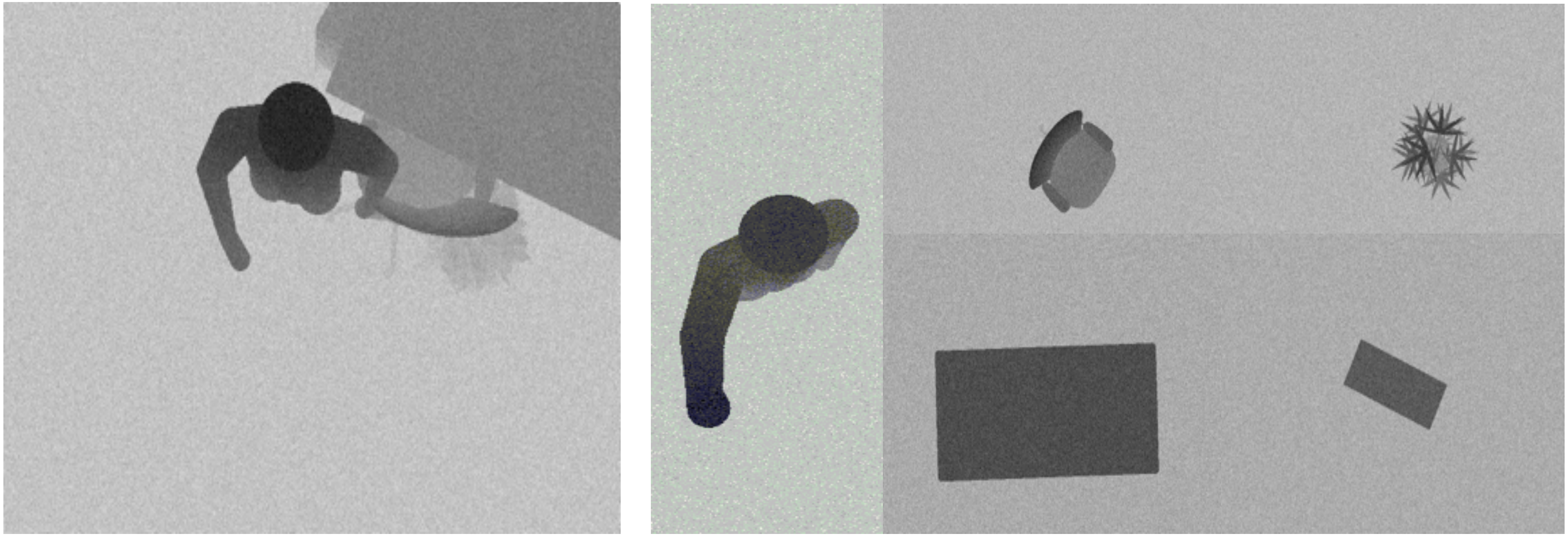


Figure 9: Synthetic depth data generated with a snythetic KINECT sensor of plants, groundtruth(*left*) and synthetic depth frame with additive white Gaussian Noise(*right*).

Training Data Model



(a): Occluded Data (Sharma *et al.*, 2015) (b): Non Occluded Data

Figure 10: Synthetic depth data generated with a synthetic KINECT sensor of all objects, synthetic depth frame with additive white Gaussian Noise.

Scene Modeling using a Density Function

- The density function capturing the context of human-object and object-object relationships in a scene is defined as:

$$\psi(S) = \psi(H, O; \theta)\psi(O, O; \theta)$$

$$\psi(H, O; \theta) = \psi(H_h)\psi(H_p)\psi(H_{pos})\psi(H_{ori})\psi(O_h)\,\psi(O_{pos})\psi(O_{ori})\psi((H,O)_\theta)\psi((H, O)_{rel})$$

$$\psi(O, O; \theta) = \psi(O_h)\psi(O_{pos})\psi(O_{ori})\psi((O, O)_\theta)\,\psi((O, O)_{rel})$$

Notation: Scene (S), Human (H), Industrial grade-component (O), Threshold (θ), Pose (p), Height (h), Position (pos), Orientation (ori), and Relationship (rel)

Testing Data

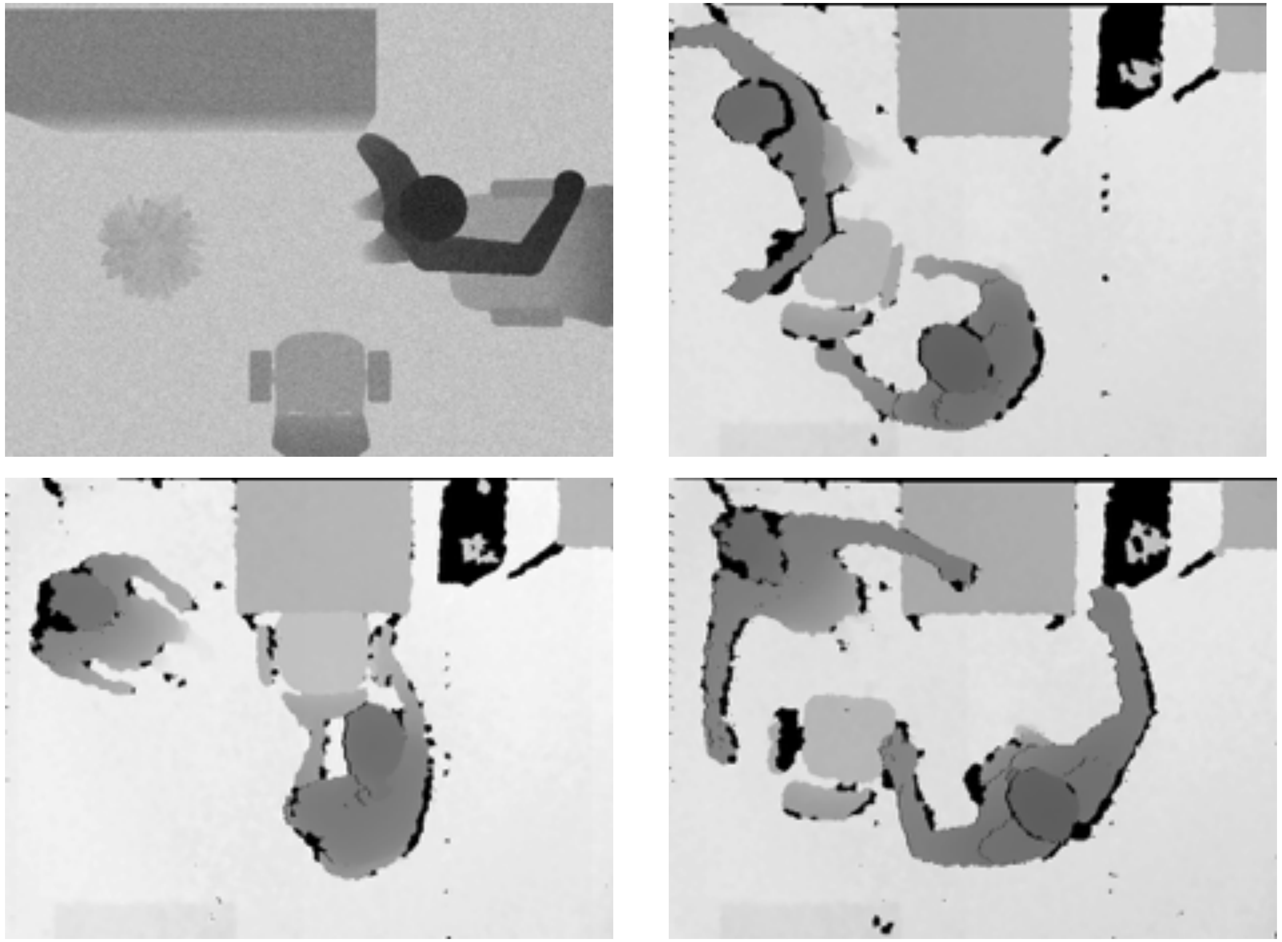


Figure 11: Real world depth data with all objects.

Selection & Definition of Feature

The features are depth information only, centered at the pixel sample patch of constant size.

Let **v** be the feature vector of the object class sample s

$$\mathbf{v}(\mathbf{s}) = (N_{[1:w],1}, N_{[1:w],2}, \ldots.., N_{[1:w],h}) \in \Re^{w.h}$$

$$N_{i,j} = d_0(s_x + (i - w/2), s_y + (j - h/2)),\ (i,j) \in \{1, \ldots., w\} \times \{1, \ldots., h\}$$

Where (s_x, s_y) is the position of sample in depth frame, $d_o(i,j)$ depicts the operator which returns the depth value at position (i,j) in the depth frame.

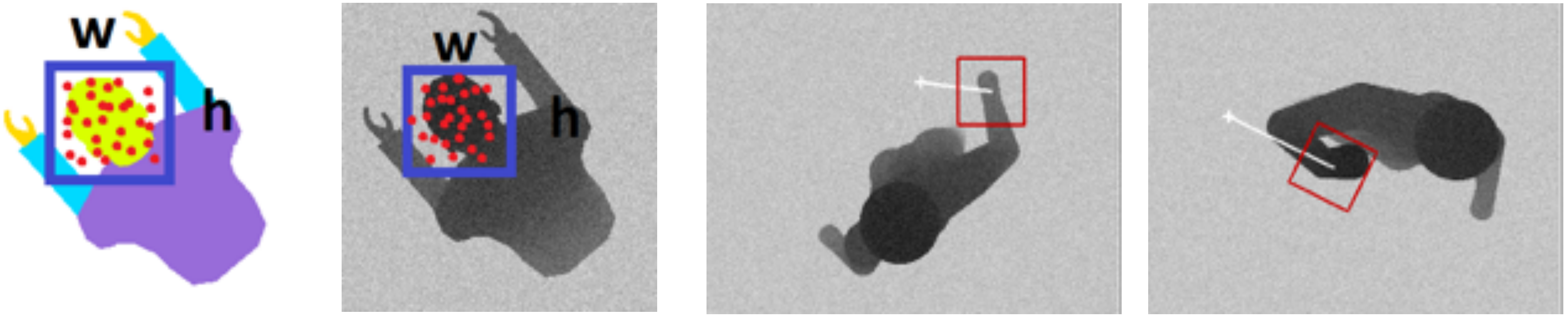


Figure 12: Feature Extraction of object class using a rectangular patch, parallel to the image coordinate system and centred at the same position.

Training and Testing Approach

- Classification Approach: Random Decision Forest (RDF) (Criminisi *et al.*, 2013)
  - Why RDF only?
    - Provides higher accuracy on previous unseen data
      - An ensemble of *n* binary decision trees is called as Forest.
      - Bagging and randomized node optimization
      - Multi-Class Classification, fast training, high generalization, easy implemetation, predictions can be understood as empirical distribution and high classification performance

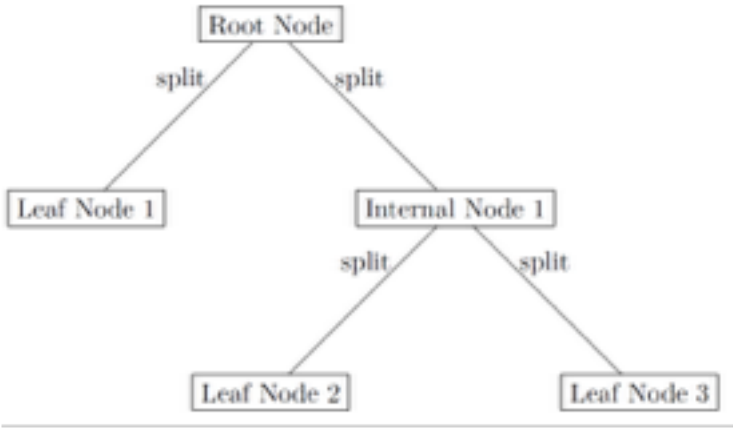
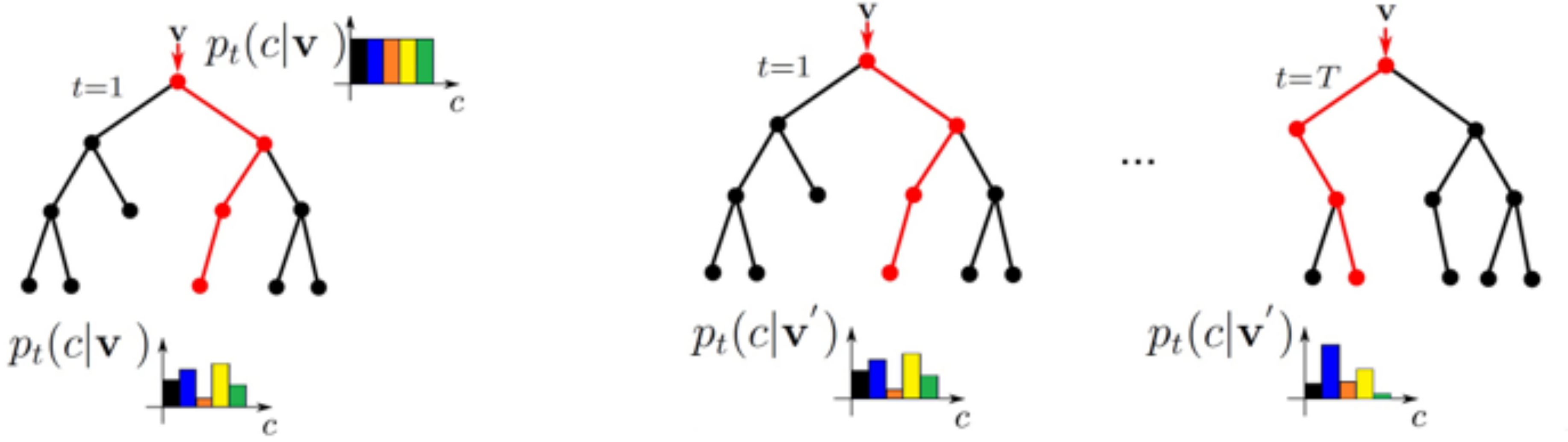


Figure 13: Structure of decision tree with root node, internal nodes and leaf nodes, along with decision criteria to split.

RDF- Training



(a): Number of decision tree =1 in a forest.

(b): Number of decision tree=T in a forest

Figure 14: RDF training with variable trees in a forest, with each tree having different dataset because of bagging.

RDF- Testing

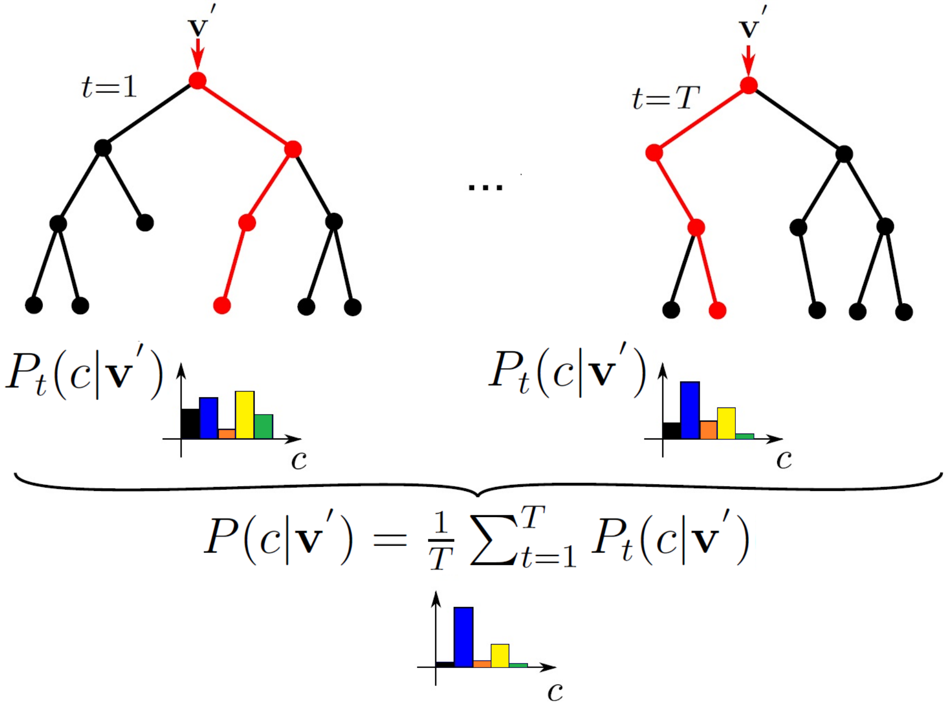


Figure 15: An example of a simple pixelwise object class labeling using RDF classifier: a query test pixel (v′) routes through each trained decision tree in a forest. Each test pixel traverses the tree through several decision nodes until it reaches the leaf node and is assigned a stored leaf statistics of the leaf node P(c|v′), where c is the class label. The forest class posterior is obtained by averaging individual tree posteriors.

Discrete Energy Minimization (CRF Extension)

- Energy Minimization methods refers to the problem of finding global minimum of a function. It is solved using α-Expansion built on Graph Cuts (Boykov *et al.*, 2001).
- Assign a label from a discrete set of labels to each pixel in an image, the models are modeled on pairwise CRF and are natually formulated as Energy Minimization Problem.
- The labeling(**x**) one aims to find a label assignment to a pixel which minimizes the energy and gives the most optimal labeling, defined as

$$E(\mathbf{x}) = E_{data}(\mathbf{x}) + E_{smooth}(\mathbf{x})$$

$$E(\mathbf{x}) = \sum_{i \in \upsilon} \varphi_i(x_i) + \sum_{i \in \upsilon, j \in \eta} \varphi_{i,j}(x_i, x_j)$$

Where ν *is the vertex (or node or pixel)* and η *is the neighbouring vertices.*

Evaluation

- For the evaluation of the overall segmentation approach, we use fixed parameter setup with
  - Forest size **T = 5**
  - Fixed patch size **(w,h) = (64,64)**
  - Maximum tree depth **D = 19**
  - For the randomization (**Ro**) in the training process **100 thresholds** and **100 feaure functions**
  - Training is based on synthetic depth frames with additive white Gaussian **noise** using a std of **15 cm**
  - In total 5000 depth frames were generated , **1600 depth frames** **(F)** were chosen in random for training (**Data**), **300 pixel positions per object class** **(PC)** were chosen uniform in random.
- PC with Intel i7 CPU with **4 core** processor**, 250GB** SSD and 4 GB RAM, pixel prediction for a frame with 640 X 480 pixels.

Quality Measure

- **Confusion Matrix**

Data Class	Relevant	Not Relevant
Retrieved	tp	fp
Not Retrieved	fn	tn

- **Precision**: Fraction of retrieved pixel based class labels, that are relevant to the actual object class labels. Mean average precision (mAP).
- **Recall**: Fraction of relevant object class labels in segmentation that are retrieved. Mean average recall (mAR).

Number of Frames and Tree Depth

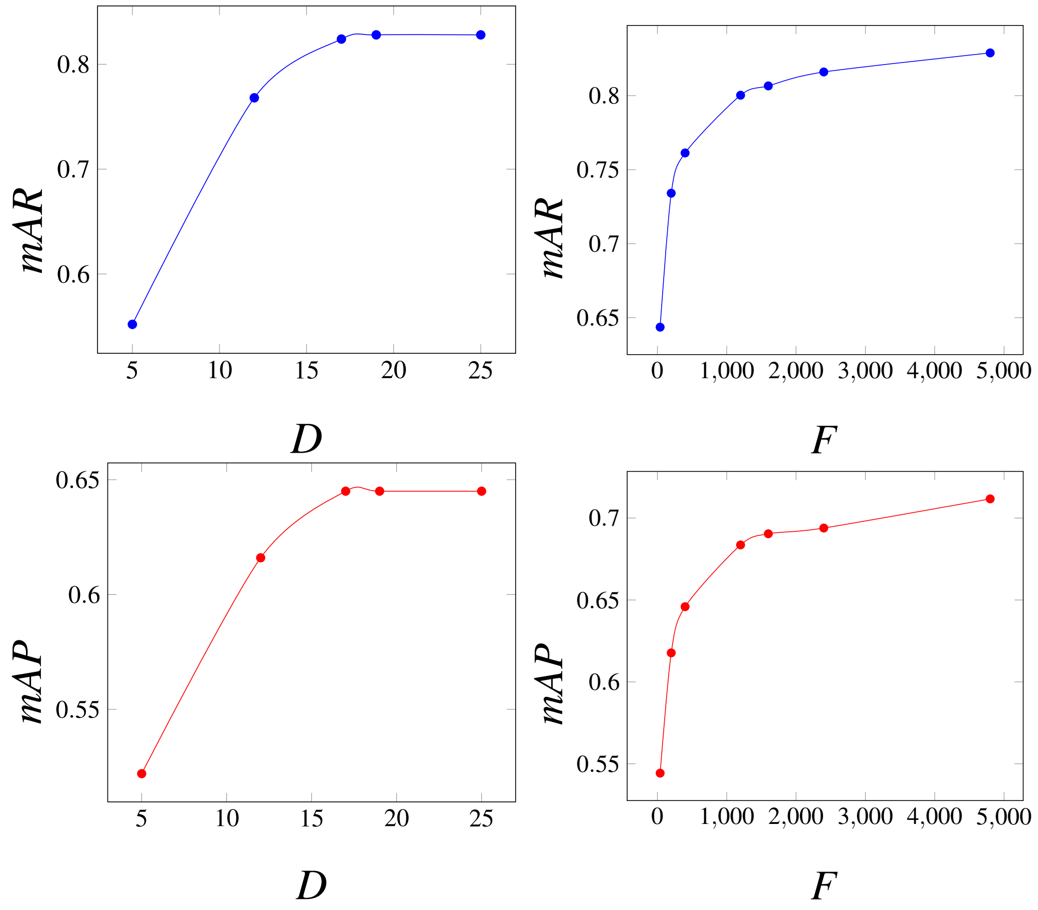


Figure 16: Confusion matrix based quality measures over an average of 65 synthetic testing frames, for variable # of training frames and tree depth

Training Time

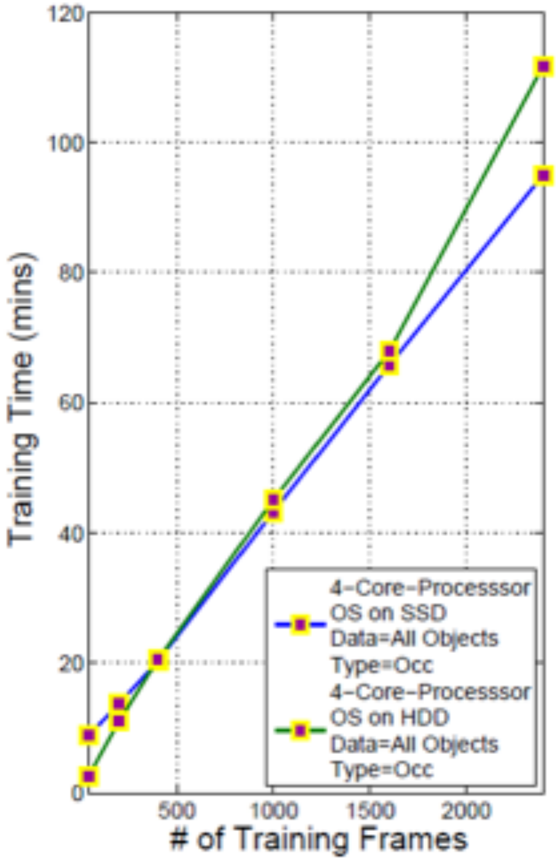


Figure 17: Training time of RDF classification tree based on # of synthetic testing frames, with type occluded data with all objects.

Comparison between RDF and CRF Extension predictions.

- **Training Data** = Synthetic depth data with all object classes
- **Testing Data** = Real world Data
- **Fixed Parameters**
  - F=1600
  - PC=300 (pixel positions per object class)
  - D=19
  - T=5
  - Ro=200 (i.e. feature function sample count=100 and thresholds=100)
  - Feat=Linear

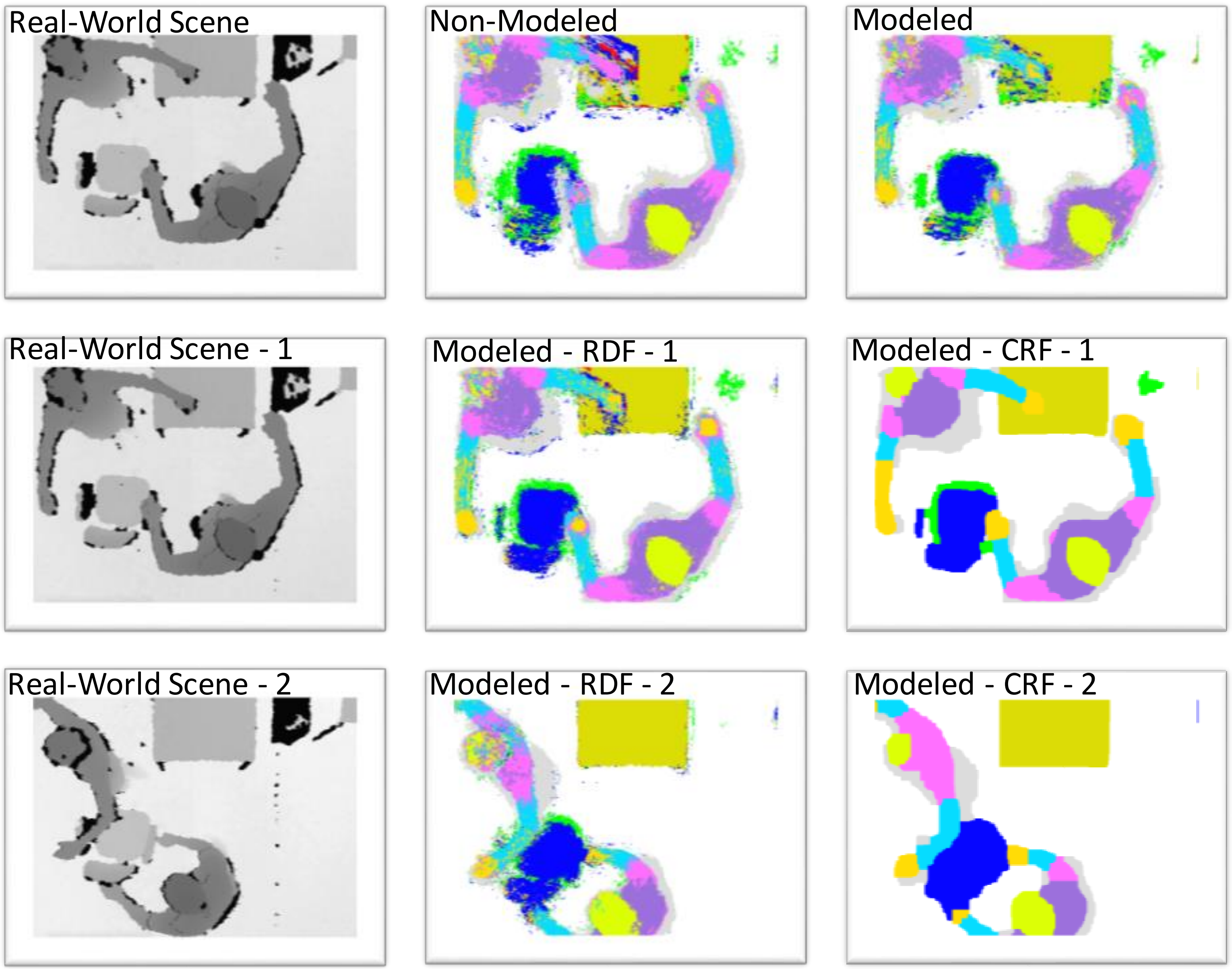


Figure 18: Prediction results based on synthetic and real- world test depth data. The first row is based on synthetic test data, the second and third rows are based on real-world test data.

Table 1: Confusion matrix based mean average recall, precision, and F1-measures

	Avg	Head	Body	UArm	LArm	Hand	Legs	Chair	Plant	Storage	Table
SOA$-RDF_{mAR}$	**0.816**	0.931	0.795	0.718	0.612	0.699	0.972	0.705	0.970	0.930	0.930
SOA$-RDF_{mAP}$	**0.620**	0.971	0.632	0.718	0.709	0.639	0.238	0.941	0.413	0.948	0.948
Ours$-CRFextension_{mAR}$	**0.885**	0.946	0.835	0.849	0.651	0.791	0.987	0.960	0.974	1.0	1.0
Ours$-CRFextension_{mAP}$	**0.819**	0.975	0.849	0.741	0.777	0.802	0.361	0.919	0.846	0.977	0.944
SOA$-RDF_{F1-measure}$	**0.734**	0.950	0.704	0.718	0.656	0.667	0.382	0.806	0.579	0.938	0.938
Ours$-CRF_{F1-measure}$	**0.842**	0.960	0.841	0.791	0.708	0.796	0.528	0.939	0.905	0.988	0.971

Comparison with SOA

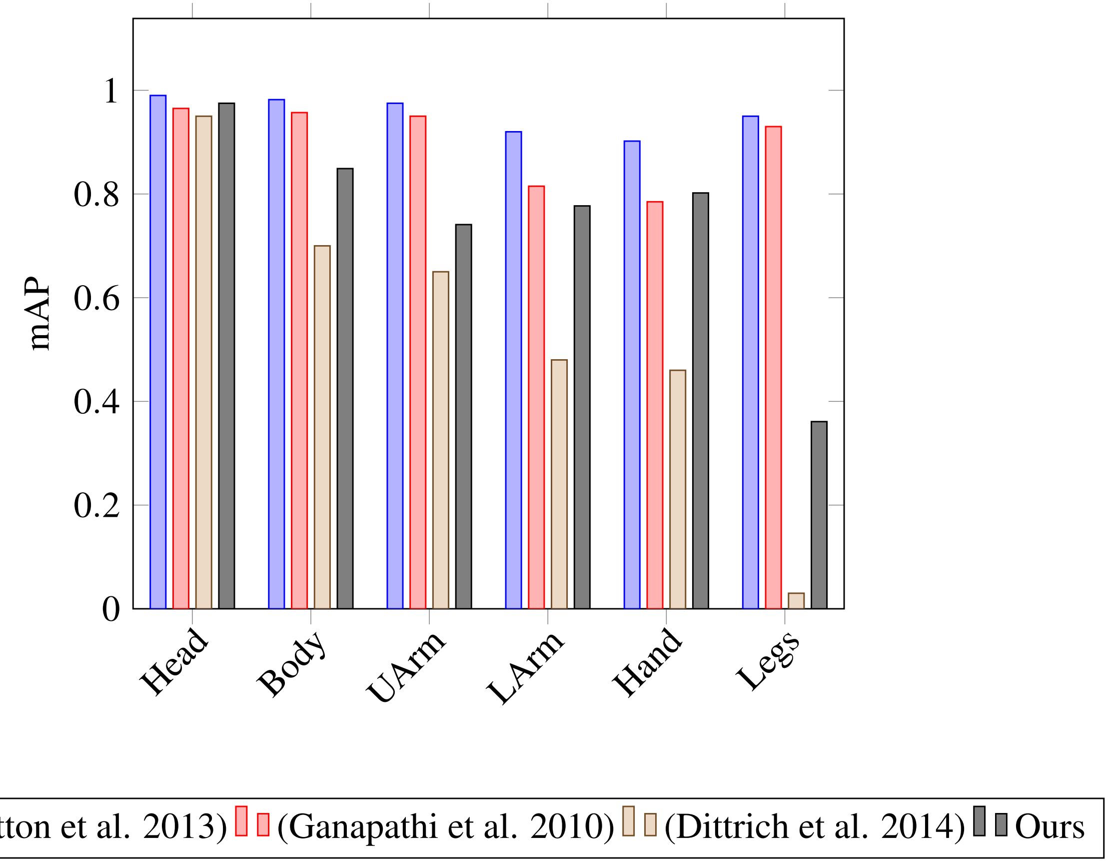


Figure 19: Comparison with (Shotton *et al.*, 2013), (Ganapathi *et al.*, 2010) and (Dittrich *et al.*, 2014). Our approach is sufficient for producing almost comparable results for localizing the joints of the human body-parts. Setups are different.

Conclusion

- We propose a generic classification for pixelwise object class labeling framework.
- The work is applied to real-time labeling (or segmentation) in RGB-D data from a KINECT sensor mounted on a ceiling placed at the height of 3.5 meters.
- The CRF extension improves the performance measures by approximately 6.9% in mAR, 19.9% in mAP, and 10.8% in F1-measure over the RDF performance measures.
- In (Shotton *et al.*, 2013), the authors “*fail to distinguish subtle changes in the depth image such as crossed arms*”, this is solved by using our training dataset based on “*top-view*”.

References

Vivek Sharma, Frank Dittrich, Sule Yayilgan and Luc Van Gool (2015). Improving Human Pose Recognition Accuracy using CRF Modeling. In *CVPR Workshops*.

Vivek Sharma, Sule Yayilgan, and Luc Van Gool (2015). Scene Modeling using a Density Function Improves Segmentation Performance. *KU Leuven, Technical Report*.

Frank Dittrich, Vivek Sharma, Heinz Woern, and Sule Yayilgan (2014). Pixelwise object class segmentation based on synthetic data using an optimized training strategy. In *ICNSC*.

Antonio Criminisi and Jamie Shotton (2013). Decision Forests for Computer Vision and Medical Image Analysis. *Advances in Computer Vision and Patter Recognition (ACVPR).*

Marc Freese, Surya P. N. Singh, Fumio Ozaki, and Nobuto Matsuhira (2010).
Virtual robot experimentation platform v-rep: A versatile 3d robot simulator.
In *SIMPAR*.

Yuri Boykov, Olga Veksler, and Ramin Zabih (2001).
Fast approximate energy minimization via graph cuts. *IEEE Trans. PAMI*.

Jamie Shotton, Ross B. Girshick, Andrew W. Fitzgibbon, Toby Sharp, Mat Cook, Mark Finocchio, Richard Moore, Pushmeet Kohli, Antonio Criminisi, Alex Kipman, and Andrew Blake (2013).
Efficient human pose estimation from single depth images. *IEEE Trans. PAMI*.

Varun Ganapathi, Christian Plagemann, Daphne Koller, and Sebastian Thrun (2010). Real time motion capture using a single time-of-flight camera. In *CVPR*.

Thanks :)