How to tune a Random Forest for Real-Time Segmentation in Safe Human-Robot Collaboration?

Vivek Sharma[1,2,3], Frank Dittrich[2], Şule Yildirim Yayilgan[1], Ali Shariq Imran[1], and Heinz Wörn[2]

[1] ESAT-PSI, Center for Processing Speech and Images,
iMinds, University of Leuven, Belgium
{vivek.sharma}@esat.kuleuven.be
[2] Faculty of Computer Science and Media Technology,
Gjøvik University College, Norway
{sule.yayilgan,ali.imran}@hig.no
[3] Institute for Process Control and Robotics,
Karlsruhe Institute of Technology, Germany
{frank.dittrich,woern}@kit.edu

Abstract. This paper is an extension of our work related to a generic classification approach for low-level human body-parts segmentation in RGB-D data. In this paper, we discuss the impact of decision tree parameters, number of training frames and pixel count per object-class during a random forests classifier training. From the evaluation, we observed that a varied non-redundant training samples makes the decision tree learn the most. Pixel count per object-class should be just adequate otherwise it may lead to under/over-fitting problem. We found a highly optimized and a most optimal parameter setup for a random forests classifier training. Our new dataset of RGB-D data of human body-parts and industrial-grade components is publicly available for lease for academic and research purposes.

Keywords: Safe Human-Robot Interaction, Random Decision Forest, Parameter Optimization, Image Processing, Object Segmentation

1 Introduction

Interest in robotics in the domain of manufacturing industry has shown an outstanding growth recently in scenarios where human beings are present too. Humans and robots often share the same workspace posing great threats to safety issues [1]. In this paper, we use a random decision forests (RDF) for pixelwise segmentation of human body-parts and industrial-grade components in RGB-D sensor data with intended use for the safe human-robot collaboration (SHRC) and interaction (SHRI) in challenging industrial environment. The major advantage of choosing an RDF approach over the classical Support Vector Machine (SVM) approach and boosting is that the RDF can handle both binary and

multi-class problems even with the same classification model. The goal of our work is to do high quality segmentation in real-time and this directly depends on the classifier parameters. In the experimental evaluation we discuss how to tune manually the training parameters[1] of the RDF and also investigate how they could be optimized for the real-time object-class segmentation time with high performance (mean average recall (mAR) and mean average precision (mAP)) scores in the evaluation.

2 Related Work

In [3], Shotton et al. inform that in their segmentation approch, most of the improvement in the performance is due to the pixelwise classification. In [3], the authors use boosted classifier for the segmentation task. However, we use a random forests classifier [2] for the segmentation task. The reasons for choosing to use RDF classifier rather than modeling the system with conditional random fields can be further found in [2, 4, 6].

Our approach is driven by three key objectives: computational efficiency, robustness and time efficiency (i.e. real-time). Our basic design of the system differs from [5] in the following aspects. In [5], all training data of human were thereby synthetically generated by applying motion capture, while we use a simple synthetic human body representation in a virtual environment using the KINECT skeleton estimations [2] which reduces the computational expense. In [5], training object samples are simple feature vectors whereas we have an optimized training strategy [2] with a reduced number of training samples while preserving the classification performance. This in turn reduces the computational expense. In [5], the authors use F=300K/tree with PC=2000 which takes a lot of training time, has a high computational cost and has large memory consumption, while in our case F=1600/tree with PC=300 is sufficient for producing almost comparable results, hence reducing computational expense and training-time. Also in [5], the authors "*fail to distinguish subtle changes in the depth image such as crossed arms*".

3 Experimental Evaluation

Each of the parameters are tuned one by one and their effects are investigated for the evaluation of 65 real-world (Real), synthetic (Syn), and test depth frames (Data), where each frame generated from a KINECT camera was of size 640×480 pixels. A random forests classifier predicts a likelihood probability of a pixel belonging to an object-class. For showing the qualitative results, if the prediction of an object-class label assigned to a pixel is less than the probability thresholding of 0.4, then color black is shown for those pixels else object-class label is shown.

[1] Additive White Gaussian Noise (σ), Tree Depth (**D**), Number of Trees (**T**), Randomization Parameter (**Ro**), Number of Training Frames (**F**), Pixel Count per Object-Class (**PC**)

Number of Training Frames: F=1600 was chosen as the most optimal value for this parameter. It was found that that an the increase in number of training frames monotonically increases the testing prediction only if the training set is highly varied i.e. redundancy in training samples does not lead the decision forest to learn more, but the confidence (precision) increases at the expense of recall (see Figure 1-2).

Pixel Count per Object-Class: As the PC[2] is increased, the RDF classifier is able to use more spatial context to make decisions, nevertheless non availability of enough pixel count would ultimately risk over-fitting to this context. Though increasing pixel count adds extra run-time cost, but improves the classification results way better. In our case most of the gains happened for the case with F=1600 and PC=300 (see Figure 3-4).

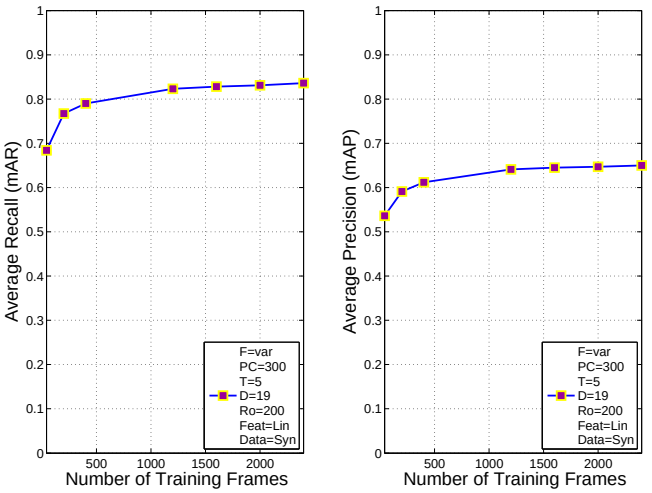

Fig. 1. Effect of the number of training frames on average recall and precision measures of pixelwise object class segmentation.

4 Conclusion

A highly optimized fixed parameters setup (i.e. F=1600, PC=300, D=19, T=5, Ro=200 and σ=15 cm) resulted in approx. 2.076×10^6 synthetic labeled training samples per tree, with a training time of approx. 43 mins. Calculating the pixelwise predictions using the trained forest takes 34 ms on a desktop with Intel

[2] Pixels extracted from a patch with features specific to a particular object-class

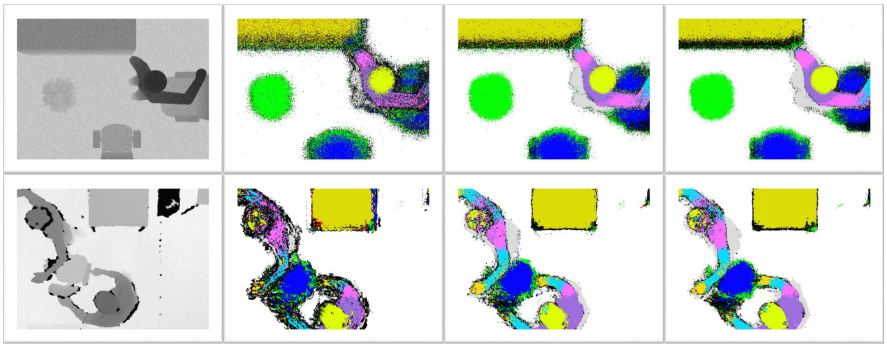

Fig. 2. Prediction results based on synthetic and real-world test depth data for number of training frames F={40, 1600, 2400}. The first column shows the test depth data, and second, third, fourth columns show the corresponding prediction results respectively for F={40, 1600, 2400} with probability thresholding of 0.4. Class predictions with a probability less than the thresholds are colored black in the result images.

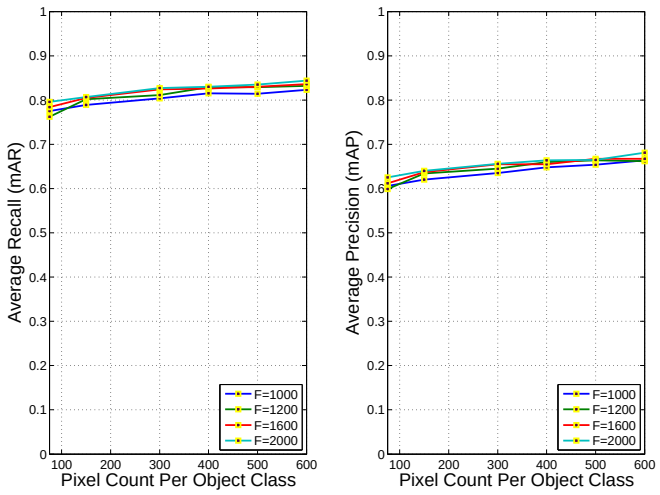

Fig. 3. Effect of the pixel count per object class on average recall and precision measures of pixelwise object class segmentation.

i7-2600K CPU at 3.40GHZ. It was demonstrated that the developed approach is robust and well adapted to the application targeted for real-time object-class segmentation in the industrial domain with humans and industrial-grade components, with mAR=0.891 and mAP=0.809. Our work can distinguish subtle changes such as crossed-arms, which is not possible in [5].

Besides, a new dataset of pixelwise RGB-D data of human body-parts composed of frames from "*top-view*" has been established. In the dataset, the human

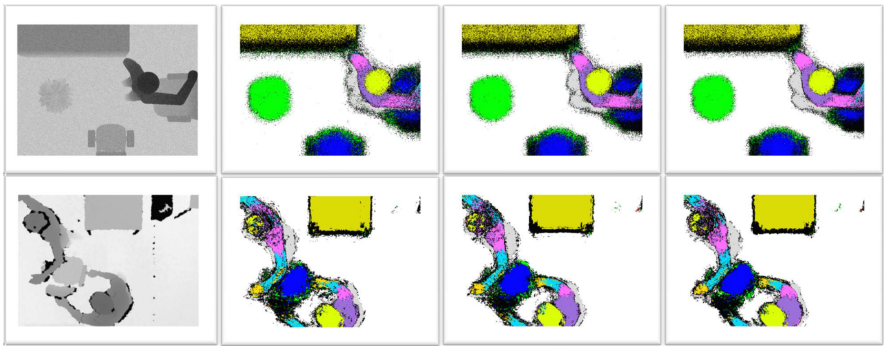

Fig. 4. Prediction results based on synthetic and real-world test depth data for pixel count per object class PC={75, 300, 600}. The first column shows the test depth data, and second, third, fourth columns show the corresponding prediction results respectively for PC={75, 300, 600} with probability thresholding of 0.4. Class predictions with a probability less than the thresholds are colored black in the result images.

appearance includes: *sitting, standing, walking, working, dancing, swinging, boxing, tilting, bending, bowing, and stretching* with combinations of angled arms, single and both arms and other combinations. Human height range is between 160-190 cm.

Acknowledgements: This work is supported by the BMBF funded project AMIKA and the EU project ROVINA.

References

1. Fraunhofer Institute for Factory Operation and Automation IFF `http://www.iff.fraunhofer.de/en/business-units/robotic-systems/research/human-robot-interaction.html`, Nov. 2014.
2. Frank Dittrich, Vivek Sharma, Heinz Wörn and Sule Yildirim-Yayilgan. Pixelwise Object Class Segmentation based on Synthetic Data using an Optimized Training Strategy. *IEEE Intl. Conf. on Networks & Soft Computing*, 2014.
3. Jamie Shotton, John Winn, Carsten Rother, and Antonio Criminisi.Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. *Int. J. Comput. Vision*, 2009.
4. Antonio Criminisi and Jamie Shotton. Decision Forests for Computer Vision and Medical Image Analysis.*Springer Publishing Company, Incorporated*, 2013.
5. Jamie Shotton, Ross Girshick, Andrew Fitzgibbon, Toby Sharp, Mat Cook, Mark Finocchio, Richard Moore, Pushmeet Kohli, Antonio Criminisi, Alex Kipman, and Andrew Blake. Efficient human pose estimation from single depth images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2013
6. Vivek Sharma. Training and Evaluation of a Framework for Pixel-Wise Object Class Segmentation based on Synthetic Depth Data. Master Thesis. *Karlsruhe Institute of Technology, University of Oslo, Hospital of Oslo and Gjøvik University College*, 2014.