RT Lions Team Description Paper

Prof. Dr. Matthias Rätsch, Thomas Weber, Sergey Triputen, Marvin Ott, Peter Stengl, Pengfei Huyan, Bo Zhang, He Lin, Steffen Eißler, Michael Litz, Moritz Mähr, Lennart Kraft, Le Ping Peng, Päivi Kärnä, Patrik Huber, and Michael Danner

RT Lions
Reutlingen University
Alteburgstraße 150
72762 Reutlingen, Germany
teammail@rt-lions.de
`http://www.rt-lions.de`
`https://www.visir.org`

Abstract. Our main research focus is image understanding. We aim to enhance human robot collaboration with our research in face detection and analysis, and object detection and tracking in many forms. Our research is applied in, including but not limited to, *Face Models for Face Recognition*, *ACI* and CNN and LSD-SLAM fusion techniques.

1 Introducing the Team

This paper describes the RoboCup@Home team **RT Lions** of Reutlingen University, Germany. It is part of the application for participation in RoboCup@Home 2019 in Sydney, Australia.

The **RT Lions** were founded in 2009. Since then our team has carried out numerous projects and researches in various fields. As the team has done a wide range of researches, it is not an easy task to break it down to only the most important ones. Research explained here is focusing on the most recent work. For further information please referrer to our website `https://www.visir.org/`.

Our team consists primarily of students. Some of us partake in scientific research and others just for fun and experience in our spare time. We have a quite varied and diverse team in terms of experience and expertise. Members range from students of the first semester in Bachelor's degree up to PhD students. They also stem form multiple faculties of our university. Nevertheless, every one of us has tasks of their own, considering both ability and interest. Our main platform is a *MetraLabs* SCITOS G5, named `Leonie`. The newest addition to our team is `LeonaRT`, a brand new robot built in summer 2018 combining a *Rethink Robotics Sawyer* collaborative robot arm and a *Neobotix* MPO-700 mobile platform.

We are happy to name companies like *Cognitec, Bosch* and *Daimler* as our industrial partners. Furthermore, we have several industrial contracts and funding programs. So far we have published more than 50 journals and conference publications. We have also accumulated a decent amount of experience from participating in competition and past victories.

We were (1) World Champion 2009 in Graz, Austria, (2) German Master in 2009, (3) Vice World Champion 2010 in Singapore, (4) Iran Master 2011, (5) 1^{st} place Informatics Inside 2014 and 2015, (6) 1^{st} place Freebots League at XVI. Portuguese Robotics Open 2016 in Branganca, (7) took part up to stage II, prepending the finals, at RoboCup@Home 2017 in Nagoya, Japan (8) and recently 1^{st} place Sick Robot Day 2018 in Waldkirch, Germany.

2 Research

2.1 Face Models for Face Recognition

Face Modeling for Pose Normalization Pose normalization is reviewed as a method to enhance the preprocessing part of a face recognition system. The orientation of an object in a three-dimensional space can be described by using the Euler Angles roll, pitch and yaw. A frontal image of a face has no rotation components relative to the camera coordinate system. Per convention, the Euler Angles in this normalized case are zero. Pose Normalization aims to reconstruct such frontal representations even with non-frontal images by the use of algorithms. One approach for obtaining a pose normalized, frontal face image is through fitting a 3D face model to the 2D input image. In the fitting process, the orientation of the input image's face is applied to the model and, depending on the method, also an optimization of shape and texture characteristics.

The next step is setting the angles of this 3D model head to zero by turning the 3D model to the normalized position. The fitted model, which is now frontal, can then be rendered back to a 2D image, which can be used by the face recognition engine.

3D Morphable Model The 3D morphable model (3DMM) is a face modelling technique that can be used for quickly generating 3D models of faces. It was first proposed by Thomas Vetter and Volker Blanz at the University of Tübingen. Various publications have been made in this field, describing new methods for obtaining a 3D morphable model, although the basic concepts remain similar.

The 3DMM consists of three-dimensional meshes of real face scans that have been registered to a reference mesh and a texture map. The initial version of this model represents the average face out of the real faces that have been used for the generation of it. Deviations of this reference mesh, novel unique faces, can be calculated by changing the shape and texture parameters of the model [1].

Mathematically, the resulting face is a linear combination of the registered real faces. Parameters of a 3DMM include the model parameters for shape, α and texture, β, and a set of projection parameters, ρ. The projection parameters include 3D rotations and translations, and the focal length of a virtual camera.

Fitting The morphable model can be used to reconstruct a 3D representation from a 2D image through fitting. Fitting is the process of adapting the 3D morphable model in such a way, that the difference to a 2D input image is minimal. Given an input image, I and a set of facial landmarks, an initial guess of the model's parameters is calculated. Then, a cost function iteratively minimizes the difference between the modeled image, I_m and the original input image, I by adapting the shape parameters, α and β, and the projection parameters, ρ for each model vertice, k:

$$\alpha, \beta, \rho = \arg \min_{\alpha,\beta,\rho} \sum_{\forall k} ||I_m(k; \alpha, \beta, \rho) - I(x_k, y_k)||^2 \tag{1}$$

This method is called analysis by synthesis, because in each iteration, a new fitting is produced and compared with the original image until the difference between model and original input is sufficiently small. After fitting, various conditions can be changed, for example, the illumination and pose of the face model. The software is open source and available on `https://github.com/patrikhuber`.

2.2 Active Closer Inspection

Face recognition For many applications face recognition plays an increasingly important role. The analysis of mimic, age, gender and movement of face actions are necessary. High resolution images are required for good quality face analysis. Furthermore, it is important that this camera system can cover a range as wide as possible with the least movement of the camera.

Since we do not have a camera with wide viewing angles and a very high resolution, a system with two specific cameras and a pan-tilt-unit (PTU) is necessary. A wide-angle camera should scan the surrounding for faces. When it finds a face, the PTU will pan and tilt a zoom camera to the target. For recognition, it may be essential to zoom in to the face. Figure 1 shows our full pipeline.

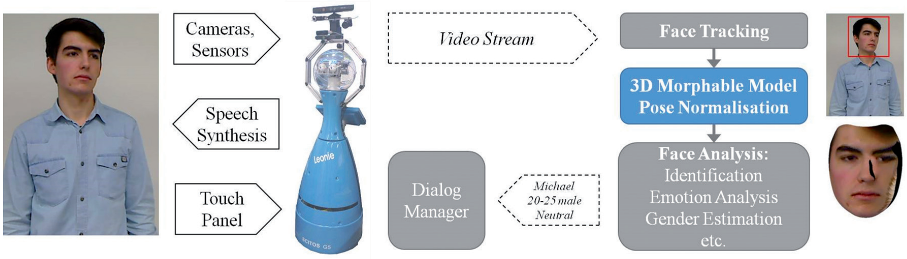


Fig. 1: Pipeline using face recognition and tracking

Single Camera Tracking The simplest tracking system requires only one camera. To track a larger area the camera is mounted on a pan-tilt-unit. For a more accurate recognition of faces with a large distance to the camera system we use a zoom camera for better results for face recognition.

The advantage is we only need a simple control which checks the eccentricity of the face in image and adjusts the angles of the PTU. The disadvantage is that the movements of the PTU are necessary to include all areas around `Leonie`.

Dual Camera Tracking To compensate the disadvantage of the single camera tracking, we need a second special camera. This this a wide angle camera which covers a large area. It is mount statically and is independent of the PTU camera movements. Thus we can scan a huge tracking area without PTU movements, but the resolution of the wide angle cameras is insufficient to recognize faces credibly. We have to convert the pixel-applied position information of the static wide angle camera to pan and tilt angles for the PTU and a zoom level for the zoom camera.

Combined Dual Cam Tracking This tracking system combines the advantages of both systems above for effectual results [2, 3]. We use two cameras in the same build-up as dual camera tracking. For face initialization, we need dual cam tracking to move PTU at the right position, for further tasks after detection and conversion, we use the features of single camera tracking. Therefore we can track faces even more accurately in comparison to the simple dual cam tracking. So this system needs two tracking: Once on the wide angle camera for the time between the first face detection and recovering on PTU zoom camera and a second tracking on PTU zoom camera after recovering. While tracking a face on PTU zoom camera, we need the tracking on wide angle camera to detect other faces in the surroundings. The disadvantages in relation to the other systems are that this system needs the most programming expense and process resources.

2.3 Follow Me: Real-Time in the Wild Person Tracking Application for Autonomous Robotics

We built a user friendly setup, which enables a person to lead the robot in an unknown environment. The environment has to be perceived by means of sensory input. For realizing a cost and resource efficient Follow Me application we need only two sensors: A monocular camera and an inertial measurement unit (IMU). With the camera input we detect and track a person. We achieve this by combining state of the art deep learning with Convolutional Neural Network (CNN) and Simultaneous Localization and Mapping (SLAM) algorithms functionality on the same input camera image. Based on the output, face recognition and robot navigation is possible. This workflow is shown in fig. 2. Our application's delivered point clouds are also used for surface reconstruction. For demonstration we use our platform SCITOS G5 equipped with the afore mentioned sensors. [4]

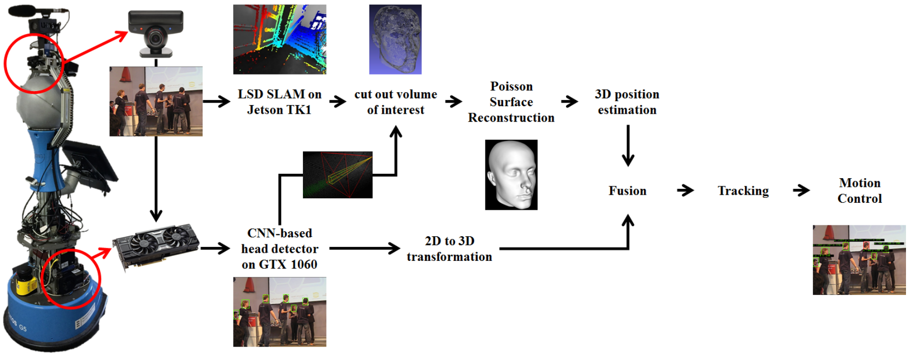


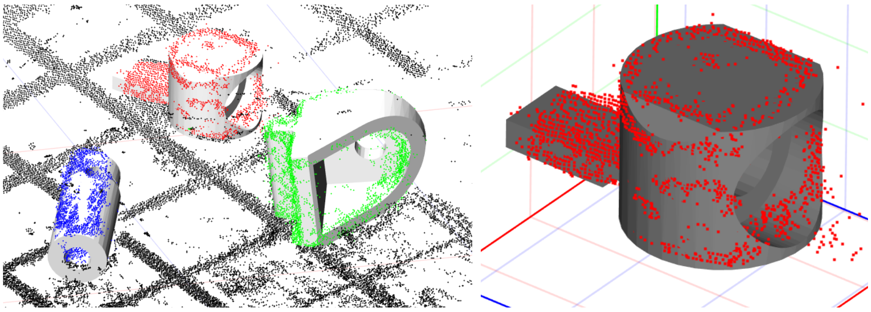
Fig. 2: Workflow of *Follow Me*: 2D image used for parallel CNN head detection and LSD-SLAM point cloud generation. Point cloud reduced to volume of head based on CNN bounding box. 3D surface reconstruction and 2D transformation for head world space coordinates. Fusion of SLAM and CNN head position. Tracking and motion control.

2.4 3D Object Reconstruction based on CNN Semantic Segmentation for Efficient and Robust Monocular SLAM

SLAM technology is implemented for various applications, especially in the field of robot navigation. We show the advantage of SLAM technology for independent 3D object reconstruction. To receive a point cloud of every object of interest void of its environment, we leverage deep learning. We utilize recent CNN deep learning research for accurate semantic segmentation of objects. We propose two fusion methods for robust CNN-based semantic segmentation and SLAM for the 3D reconstruction of the objects of interest in order to obtain a more robust and efficient 3D reconstructions. As a major novelty, we introduce a CNN-based masking to focus SLAM only on feature points belonging to every single object. Noisy, complex or even non-rigid features in the background are filtered out, improving the estimation of the camera pose and the 3D point cloud of each object, as shown in fig. 3. We analyze the accuracy and performance of each method and compare the two methods describing their pros and cons. [5]

3 Hardware

Robot `LeonaRT` We built `LeonaRT` in summer 2018. It is a combination of a *Neobotix* MPO-700 omnidirectional mobile platform and a *Rethink Robotics Sawyer* collaborative robot arm. Both components are fully integrated into a new system, with the Robot Operating System (ROS) as the common software foundation. This new system allows us to execute all sorts of new challenging tasks. A two-finger gripper combined with the force feedback of the robot arm allows gentle gripping actions and other physical interaction scenarios.



(a)

(b)

Fig. 3: (a) Point cloud produced with LSD-SLAM and the respective 3D model of the object aligned to its point cloud. (b) Point cloud of only the red object of interest.

We develop both our robots, `Leonie` and `LeonaRT`, in parallel, tho. Most of our application are in transferred onto `LeonaRT` and our ultimate goal is to start with this robot on the RoboCup@Home competition in 2019.

Robot `Leonie` For `Leonie`, we used the robot base SCITOS G5 developed by *MetraLabs Robotics*. It combines industrial and research properties in one robot. Thus it has the advantages of an industrial robot, such as robustness and longevity, and the mobility and flexibility of a research robot. With its weight of 60 kg it can move at a speed up to $1.4\,\mathrm{m\,s^{-1}}$ and handles payloads up to 50 kg without any difficulties. It contains one lithium iron phosphate battery, allowing operation up to 20 hours on normal usage.

Next Unit of Computing Two mini PC *Intel NUC Kit NUC7i7BNH* are installed on `Leonie`. With these, we practice the divide and conquer design pattern, by capsuling programs from the brain to the NUCs. We use two because of the rather large number of programmers working on `Leonie` and thus, to allow flexibility in terms of preferred operating systems.

Active Closer Inspection (ACI) To achieve a large range of visibility with sufficient details, we use two different cameras and a pan-tilt-unit (PTU). A wide-angle camera scans the surrounding for faces. If it finds a face, the PTU will pan and tilt a mounted zoom camera to the target.

Leap Motion For enhanced communication for speech and hearing impaired people, we are working with a stereo IR camera, for the purpose of hand gestures.

4 Software

B.R.A.I.N. Best Reutlingen Artificial Intelligence Network is our central control module, based on *YAKINDU Statechart Tools*. It combines deterministic finite automaton with fuzzy logic and acts as interface to all sensors and actors.

Emofani Our animated face was designed and rigged in Blender and Blend-Tree in Unity3D, by a former team member [6–8]. It simulates human like behaviour: breathing, blinking, mouth, speech animation, gaze, random eye movements and micro movements. While servo turns the head, eyes and face move back. The program is not only used in several industrial projects, but also open source available on `https://github.com/steffenwittig/emofani`.

Hand Gesture Detection We have two different projects in this topic, which are both at use on `Leonie`. The first one involves the *Leap Motion* API and is used for simple detections on `Leonie`s front. The second one uses *caffe*, a deep learning framework, for more complex detection from `Leonie`s head.

Speaker Detection This program was based on *PrimeSense*. It uses a *Microsoft Kinect 2* for detection of the angle of all bodies and its microphone array for detection of the angle towards a source of noise. In combination of both features it has a great confidence in detecting people in its view.

Speech Recognition Our speech recognition module consists of three parts: (1) The *Google Speech* API, (2) the *Stanford Parser* and (3) a decision maker. The *Google Speech* API is used to convert the input audio to text. Powered by machine learning, this API is one of the best speech recognition technologies available online. With the help of the *Stanford Parser*, we are able to tag the words in the incoming string. The tagged string is then passed onto our self-programmed decision maker, which picks out important keywords and decides what the robot should do. Such as carry out a command, answer a question or just interact with the person. For answering general knowledge questions, we integrated the *Wolfram Alpha* API into our answering-questions-program.

Navigation, Localization, Mapping, WaypointVisitor To know its environment, `Leonie` uses two laser sensors located on its front and back and an odometer. These are used to detect obstacles in its path as it moves autonomously towards its goal. The mapping is done with *MiraCenter* software. The uniqueness of the mapping implementation in *MiraCenter* is the multi-map structure. A map is split into four separate map files: (1) static, (2) nogo, (3) speed and (4) one way. Each map type has its own function enabling flexibility in controlling `Leonie`'s movements with the maps, without needing to execute special commands. Combining these two abilities, we have the WaypointVisitor enabling `Leonie` to learn multiple position and poses on a built map.

Follow Me As already briefly explained in section 2.3, we use a monocular camera system on `Leonie` to track heads. Head detection is done via a specially trained Single Shot MultiBox Detector CNN (SSD) [9]. Based on the position of the bounding box, a single object tracking based on the combination of a Kalman Filter and global nearest neighbor data assignment [10] is applied. The tracked position is used by robot navigation controller to follow the person.

Emotion Detection With this software it is possible to detect emotions of a person based on three-dimensional face landmarks. The face-landmarks are generated from the 3D landmarks in *4dface* which is using only 2D pictures.

Attractiveness Estimation Two random pictures of a database of 1578 pictures were compared by students to train the software. Combined with a *Cognitec* evaluator it is possible for the software to estimate the attractiveness of a person.

Bibliography

References

1. M. Rätsch P. Huber, J. Kittler. Bottom-up and top-down face analysis based on 3d face models. *Informatics Inside Conference for Human-Centered Computing*, pp 138, 2014.
2. P. Poschmann H.-J. Böhme M. Rätsch P. Kopp, M. Grupp. Tracking system with pose-invariant face analysis for human-robot interaction. *Informatics Inside Conference for Human-Centered Computing*, pp 70-71, 2015.
3. M. Rätsch J. Kittler H.-J. Böhme P. Poschmann, P. Huber. Fusion of tracking techniques to enhance adaptive real-time tracking of arbitrary objects. *Conference on Intelligent Human Computer Interaction (IHCI)*, DOI: 10.1016/j.procs.2014.11.025, 2014.
4. Thomas Weber, Sergey Triputen, Michael Danner, Sascha Braun, Kristiaan Schreve, and Matthias Rätsch. Follow me: Real-time in the wild person tracking application for autonomous robotics. In *RoboCup 2017: Robot World Cup XXI*, pages 156–167. Springer International Publishing, 2018.
5. Weber T. Rätsch M. Triputen, S. 3d object reconstruction based on cnn semantic segmentation for efficient and robust monocular slam. (submitted), 2018.
6. M. Rätsch S. Wittig, U. Kloos. Emotion model implementation for parameterized facial animation in human-robot-interaction. *International Conference on Computer Technology and Development (ICCTD)*, JCP, ISSN: 1796-203X, 2015.
7. U. Kloos. S. Wittig, M. Rätsch. Parameterized facial animation for socially interactive robots. *Human and Computer (MC)*, pp 355-358, 2015.
8. M. Rätsch S. Wittig, U. Kloos. Animation of parameterized facial expressions for collaborative robots. *Informatics Inside Conference for Human-Centered Computing*, pp 72-73, 2015.
9. Anguelov D. Erhan D. Szegedy C. Reed-S. Fu C.-Y. Liu, W. and A. C. Berg. Ssd: Single shot multibox detector. *ECCV*, 2016.
10. Udvarev A. Konstantinova, P. and T. Semerdjiev. A study of a targettracking algorithm using global nearest neighbor approach. *4th International Conference Conference on Computer Systems and Technologies*, CompSysTech ACM, 2003.

Robot Leonie

Hardware Description

Specification are as follows:

- Base and Torso: SCITOS G5 developed by *MetraLabs Robotics*. Combining industrial and research robot, 1.4 $\mathrm{m\,s^{-1}}$ max speed. Handles payloads of up to 50 kg.
- Contains one lithium iron phosphate battery, allowing operation up to 20 hours on normal usage.
- Two mini PC *Intel NUC Kit NUC7i7BNH*
- ACI:
  + Pan Tilt Unit *PTU-D46-17*
  + Wide Angle Camera *SVS-VISTEK SVCam-ECO*
  + Zoom Camera *Sony FCB-EV7500*
- *Microsoft Kinect 2*
- Microphone *Sennheiser MKE 600*
- Head: Mini-beamer that projects the face of Leonie on a plastic dome
- Robot dimension: height: 1.90 m, width: 0.60 m, depth: 0.70 m
- Robot weight: 80 kg

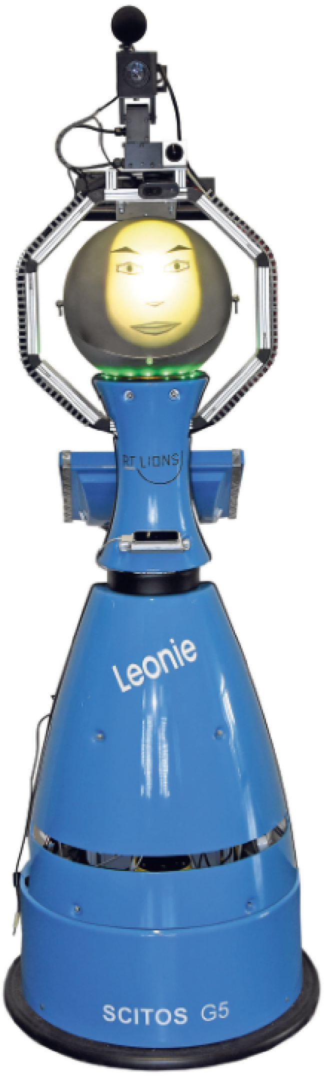


Fig. 4: Robot Leonie

Software Description

For our robot we are using the following software:

- Platform: *Windows 10* and *Ubuntu 14.04 LTS*
- Navigation, localization and mapping: *MiraCenter* and *CogniDrive*
- Speech recognition: *Google STT* and *Sphinx*
- Speech synthesis: *Ivona TTS*
- Object recognition: own software based on *caffe* framework
- Face recognition: own software based on *caffe* framework
- Other synthesis and recognition: own software

Robot LeonaRT

Hardware Description

Specifications are as follows:

- Base: *Neobotix* MPO-700 omnidirectional mobile platform, 0.9 m s^{-1} max speed. Maximum payload: 400 kg.
- Torso: Flexible assembly system profile housing containing all electric appliances.
- Contains eight lead gel batteries, allowing operation up to 10 hours on heavy usage.
- Arm: *Rethink Robotics Sawyer* collaborative robot arm mounted on torso. 7 DOF, anthropomorphic. Maximum load: 4 kg. Maximum reach: 1.26 m.
- Head: 1 DOF *Rethink Robotics Sawyer* screen displaying the face
- Additional computing device: *Dell Alienware 15 R4 Gaming (i9-8950HK / GeForce GTX 1080)*
- Robot dimensions: height: 1.50 m, width: 0.60 m, depth: 0.90 m
- Robot weight: 350 kg.

Also our robot incorporates the same of most of `Leonie`*'s sensory devices.*

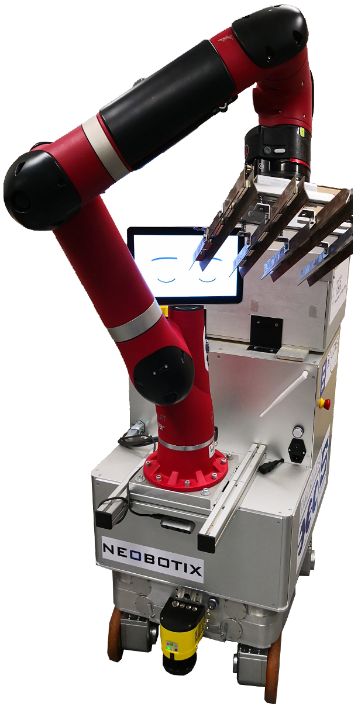


Fig. 5: Robot `LeonaRT`

Software Description

For our robot we are using the following software:

- Platform: *Ubuntu 16.04 LTS* running ROS *Kinetic* and KVMs
- Navigation, localization and mapping: ROS navigation stack with modified *teb_local_planner*
- Face recognition, speech recognition, speech synthesis, object recognition: See description on Robot `Leonie`.
- Arm control: *Rethink Robotics Intera SDK* with ROS interface