Intuitive Robot Control with a Projected Touch Interface

Lennart Claassen, Simon Aden,
Johannes Gaa, Jens Kotlarski, and Tobias Ortmaier

Institute of Mechatronic Systems,
Leibniz Universität Hannover,
Appelstraße 11a, 30167 Hanover, Germany
{lennartc,simon.aden}@stud.uni-hannover.de,
{johannes.gaa,jens.kotlarski,tobias.ortmaier}@imes.uni-hannover.de
www.imes.uni-hannover.de

Abstract. This work proposes an intuitive and adaptive interface for human machine interaction that can be used under various environmental conditions. A camera-projector-system is added to a robot manipulator allowing for a flexible determination of a suitable surface to project a graphical user interface on. The interface may then be used to select different autonomous tasks to be carried out by the robot. In combination with an implemented person tracking algorithm our approach offers an intuitive robot control, especially for repetitive tasks as they occur inside domestic or working environments.

Keywords: Robots, Intuitive Control, Projection, Touch Interface.

1 Introduction

Today's working environments of industrial production sites are still characterized by a combination of static machines and human workers. Due to safety reasons, areas where robots operate are often shielded from human access. This applies both to stationary and to mobile robotic systems. It has been shown that the joint actions of humans and robotic systems can lead to more flexibility and new possibilities [1]. In the field of service robotics the interaction of human and robotic systems became a main exploratory focus, i.e. see [2].

Previous approaches of integrating projectors into the human machine interaction have either used static robot configurations, thereby limiting the range of possible projection surfaces [3,4], or used the projectors as hand held devices to control robotic movement [5]. The objective of the presented work is the creation of an innovative user interface which allows for simple accessibility and easy operation of a robot. The application is especially designed for recurring tasks characteristic for domestic and working environments such as collecting and delivering materials or products. By projecting a graphical user interface using the robots manipulator the need for additional input equipment such as computers, mobile devices or other control panels would become redundant. Through the

M. Beetz et al. (Eds.): ICSR 2014, LNAI 8755, pp. 95–104, 2014.

flexibility offered by the robots manipulator any suitable surface in the reach of the robot may be used as a projection area for the user interface. The projected user interface then allows for selection and start of different implemented tasks. This way the proposed interface is most effective when combined with autonomous systems, which are able to carry out several tasks by their own.

An important aspect in the interaction with a mobile robot is to tell the robot where to move. In a well defined environment this can be done by choosing one of several predefined locations via the graphical interface. The additional option of manual movement control however allows more flexibility and lets the user e.g. teach new locations or direct the robot to a desired target area. For this purpose we implemented a track and follow algorithm that lets the robot track the user and follows him to any location. Combined with the freely selectable projection area, full advantage can be taken of the robots mobility.

The interface projection system and the user input detection are described in section 2. Section 3 presents the tracking system and user following algorithm. The user input detection is evaluated in section 4 followed by conclusions and possible future enhancements in section 5.

2 The Interface Projection System

To realize an intuitive control of a robot the cooperation of different components is required. These include the distortion free projection of a user interface onto a given area and the detection of the user input.

The projection system consists of two devices: a small laser projector used to project the graphical user interface and a 2D video camera to detect suitable projection areas and to capture the selection made by the user. Therefore, the first requirement for the camera-projector-system is a high degree of correlation between the camera field of view (FOV) and the projector FOV. This was achieved by creating a mounting which allowed a fixed arrangement of the two devices on the robots manipulator as shown in figure 2. The second requirement for the system is to allow for perspective transformations between the camera image frame and the projector image frame in order to transform the detected user input into the scope of the projected interface. To identify intrinsic and extrinsic parameters of the camera and the projector a calibration of the system as proposed by Raskar and Beardsley in [6] can be performed. However, since this approach makes use of external sensors a different calibration method was implemented which is based on correspondences between 3D points in the camera coordinate frame and 2D points in the projector image frame.

2.1 Camera Projector Calibration

The chosen calibration method after Leung *et al.* [7] is based on detected correspondences between the homogeneous 2D points on the projector image plane $P_\mathrm{P} = [u_\mathrm{P}, v_\mathrm{P}, 1]$ and the homogeneous 3D points inside the camera coordinate frame $P_\mathrm{C} = [x_\mathrm{C}, y_\mathrm{C}, z_\mathrm{C}, 1]$. This approach is valid because the projection model of a projector is basically the same as the model of a camera. The only difference lies in the projection direction: a camera projects 3D points into a 2D plane while a projector creates a 2D image at the intersection with the 3D points of the image plane. For this reason any known 3D point can also be projected onto the projector image plane if the projector is treated as a camera.

After identification of the intrinsic camera parameters using a camera calibration based on Zhang's method [8] the 3D point coordinates can be determined in the camera coordinate system. Afterwards a transformation of the points into the projector coordinate system would be possible given the relative rotation and translation between camera and projector. However, if this transformation is combined with the unknown intrinsic parameters of the projector to form the projection matrix M_P, the relationship between the points can directly be expressed as:

$$\begin{bmatrix} u_\mathrm{P} \\ v_\mathrm{P} \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix}}_{M_\mathrm{P}} \begin{bmatrix} x_\mathrm{C} \\ y_\mathrm{C} \\ z_\mathrm{C} \\ 1 \end{bmatrix} \quad . \tag{1}$$

Division of lines one and two and lines one and three of the equation system in (1) leads to the equations:

$$v_\mathrm{P}(x_\mathrm{C}m_{11}+y_\mathrm{C}m_{12}+z_\mathrm{C}m_{13}+m_{14})-u_\mathrm{P}(x_\mathrm{C}m_{21}+y_\mathrm{C}m_{22}+z_\mathrm{C}m_{23}+m_{24}) = 0 \quad , \tag{2}$$

$$u_\mathrm{P}(x_\mathrm{C}m_{31} + y_\mathrm{C}m_{32} + z_\mathrm{C}m_{33} + m_{34}) = (x_\mathrm{C}m_{11} + y_\mathrm{C}m_{12} + z_\mathrm{C}m_{13} + m_{14}) \quad . \tag{3}$$

By dividing the n detected correspondences into two subsets of $n_1 \geq 8$ and $n_2 \geq 4$ the equations (2) and (3) can be solved using Singular Value Decomposition to give an estimation of the projection matrix M_P.

The executed calibration now enables the transformation of any given point in 3D camera coordinates to the projector image plane.

2.2 Projection Plane Detection

After successfully calibrating the camera-projector-system the detection of a suitable projection area in the camera coordinate system is required. An area is considered suitable for projection if it is a flat plane in which a rectangular shape of at least ten centimeters in width and six centimeters in height may be fitted. The plane detection can be achieved using either the camera directly or using the forward kinematics of the robot.

Plane Detection Using the Camera. Since we are using a 2D camera the detection of a plane in 3D coordinates is only possible with at least some prior knowledge about the projection area. Using given information, such as dimension, shape or color of the plane, different image processing techniques e.g. Harris corner detection [9] or Hough transformation [10] may be applied to extract the corner points which then can be used to determine if the area is suitable for projection of the user interface.

Plane Detection Using the Forward Kinematics. If the pose of the camera has been integrated into the robot model e.g. using hand-eye calibration the forward kinematics of the robot may be used to determine a suitable projection area. In our implementation the forward kinematics is used to determine the camera pose in relation to a given plane in the robot environment, e.g. the floor plane the robot is moving on. The algorithm then searches for the largest possible projection area to fit the FOV of the projector starting from the intersection of the central projection ray and iteratively incrementing the projection area until the limitation of the projector FOV is reached.

2.3 Image Projection

After the determination of a suitable projection area and transformation of the plane into the projector image frame, the projection image has to be transformed to fit the projection area in order to be displayed to the user without any perspective distortion. This is achieved by calculating the homography matrix and applying perspective transformation to the output image as in the example shown in figure 1.

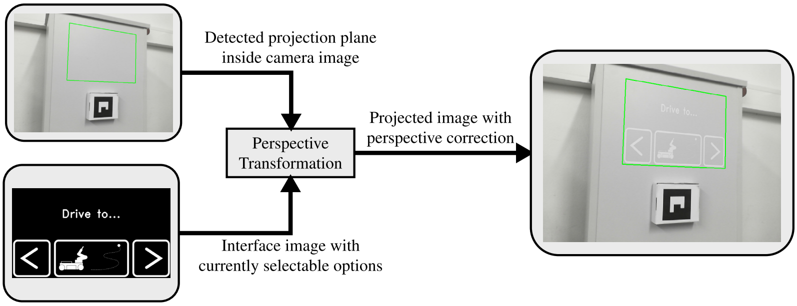


Fig. 1. Perspective transformation of the output image

2.4 Projection Area Alignment

As an example application we implemented the detection and usage of different projection stations. The stations are equipped with labels containing augmented reality (AR) code markers as well as suitable projection areas. By detecting the AR code a coordinate system can be determined inside the camera frame for every marker. Using coordinate transformations it is then possible to align the robot and the projection system to the marker and thereby to the projection surface.

Figure 2 shows the transformation used to align the robot to the projection station. First the detection of the AR marker gives the transformation ${}^{AR}T_C$ from the Coordinate System of the AR marker $(CS)_{AR}$ to the coordinate system of the camera $(CS)_C$. Using the given transformation ${}^{C}T_R$ between the camera and the robot coordinate system $(CS)_R$ from the hand-eye calibration the transformation between the AR marker and the robot can be determined as:

$$ {}^{AR}T_R = {}^{AR}T_C \, {}^{C}T_R \ . \tag{4} $$

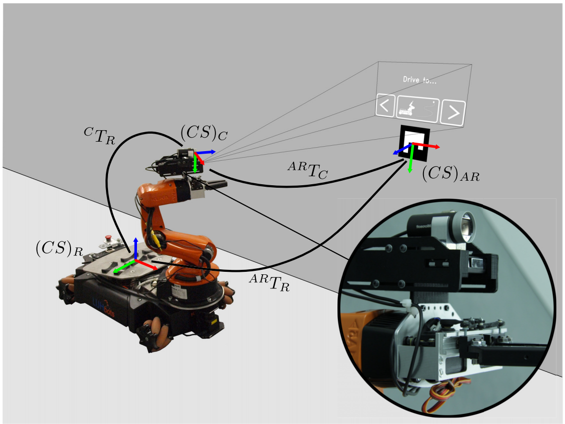


Fig. 2. Coordinate Transformation at a Projection Station

2.5 User Input Detection

As described, the detection of the user input is achieved using the 2D camera of the camera-projector-system. First, a perspective transformation of the projection area into the camera image plane is carried out, to restrict the processed section of the camera image to the projection area. It is then possible to divide the input image into sections that relate to different areas of the projected user interface. The user input e.g. touching of the projected interface buttons is then detected using the implementation of the Gaussian mixture model for background subtraction described by Zivkovic [11]. The complete process of the user input detection is shown in figure 3. By implementing an additional color filter the possible input devices may be restricted and noise in the input image can be reduced to enhance the robustness of the input detection.

3 The User Tracking System

For the task of following the user the robot is equipped with a depth camera to detect and subsequently track the person standing in front of the robot. Taking

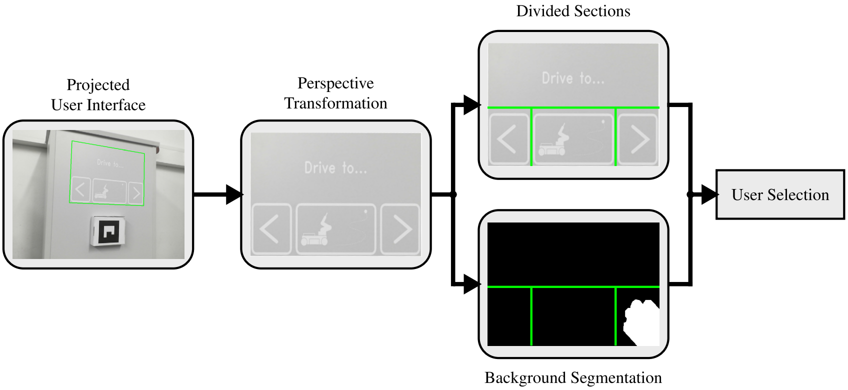


Fig. 3. User Input Detection

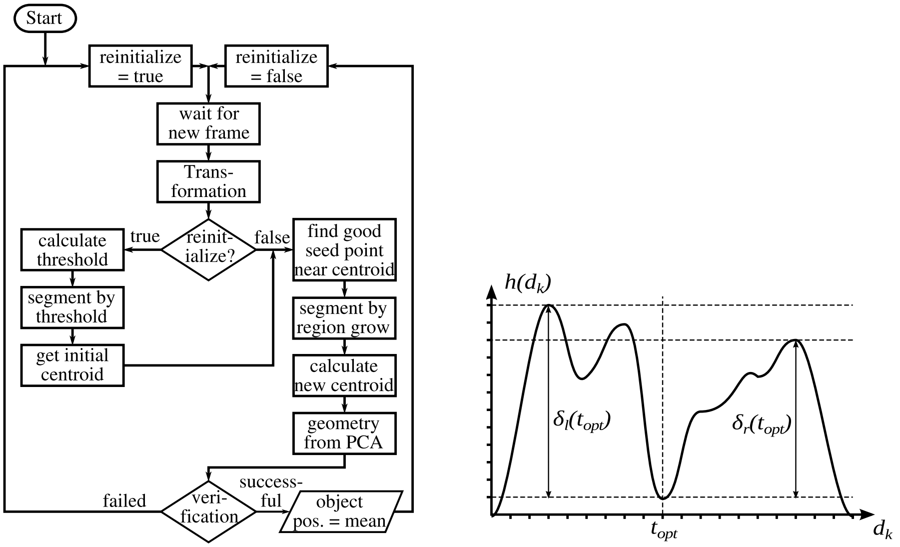
advantage of the prior knowledge of the person's position on activation we used an approach for tracking that does not depend on the detection of specific human features. This allows for a more robust detection and tracking from any camera angle and a person can be detected even if only a part of the body is visible or if the person's silhouette is unrecognizable. The retrieved position from the tracking algorithm is then used by the robot to follow the user by trying to maintain a defined distance to him. Laser scanner data is used to avoid obstacles along the way.

3.1 Tracking Algorithm

The algorithm uses the centroid of the tracked object in the previous frame as a seed point for a region growing algorithm that segments the tracked object in the current frame of the depth image. In order to verify that the correct object was found the geometric extent of the object is calculated using a principle component analysis (PCA) of the segmented point cloud. If the change of geometry is greater than a defined threshold the object is rejected and the tracker is reinitialized.

Transformation. The depth values are needed in a coordinate frame orthogonal to the tracked object. If the sensor is mounted with an angle the point cloud must be transformed to a suitable frame. The depth sensor provides a point cloud $P \in \mathbb{R}^{I \times J \times 3}$ where I and J are the height and width of the depth image and $p(i,j) \in \mathbb{R}^3$ with $i \in [0, I[$ and $j \in [0, J[$ is one Cartesian point. With the rotational matrix R the points are transformed to a coordinate frame orthogonal to the object being tracked. The transformed point cloud P_t consists of the points

$$p_t(i,j) = [x_{ij}, y_{ij}, z_{ij}]^T = R\, p(i,j) \ . \tag{5}$$



(a) Flow Chart of the Tracking Algorithm

(b) Histogram of Depth Values with Optimal Threshold

Fig. 4. Flow Chart and Depth Value Histogram of the Tracking Algorithm

Initialization. At initialization it is assumed that the object to be tracked is the dominant object in the foreground of the scene. An initial mask for the object can then be obtained by applying a threshold to the depth image.
In the histogram $h(d_k)$ of the K discretized depth values d_k the optimal threshold t_{opt} can be calculated using the following left and right distances in the histogram (also see figure 4 (b))

$$\delta_l(d_k) = \max_{i=0,\dots,k-1} (h(d_i) - h(d_k)) \ , \tag{6}$$

$$\delta_r(d_k) = \max_{i=k+1,\dots,K-1} (h(d_i) - h(d_k)) \ . \tag{7}$$

The optimal threshold maximizes the sum of both distances.

$$t_{\text{opt}} = \operatorname*{arg\,max}_{k=0,\dots K-1} (\delta_l(d_k) + \delta_r(d_k)) \ . \tag{8}$$

The centroid of all pixels with a depth value $z_{ij} < t_{\text{opt}}$ is used to find the initial seed point.

Segmentation. First a good seed point $s = [i_s, j_s]$ has to be found in the neighborhood N of the given centroid $c = [i_c, j_c]$. If the depth value of the centroid from the last frame is d_{last} then the seed point is chosen as the point with the closest depth value to d_{last}.

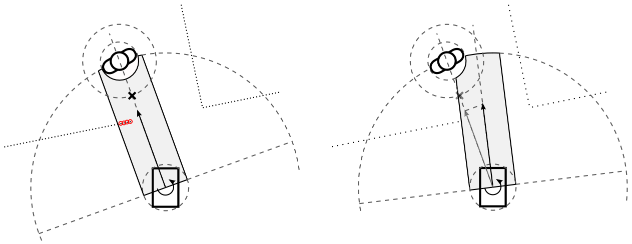
$$s = \underset{[i,j]\in N}{\arg\min}\left(|d(i,j) - d_{\text{last}}|\right) \ . \tag{9}$$

Starting from this seed point a region growing algorithm marks all connected pixels that have a depth value within a given tolerance range as foreground. Given the foreground pixels the new centroid of the object can be calculated.

Verification. Under the assumption that the geometric extent of the tracked object can not change drastically from one frame to another the consistency of the geometric properties indicates if the object has been lost. The geometric extent of the object can be estimated by a PCA which calculates the mean vector and the eigenvalues and eigenvectors of the covariance matrix of the point cloud. The decision if the detected object shall be accepted or rejected can be done by comparing the results to the ones from the previous frame.

3.2 Following Algorithm

Starting with the position of the tracked person a target point is set on the intersection point of the direct line between the robot and the person and a circle around the person's position (see figure 5 (a)). The radius of the circle defines the distance at which the robot tries to follow. The translational velocity vector is set towards the target point with an absolute value proportional to the distance. The heading of the robot is controlled towards the tracked person to ensure that the person is always within the field of view of the depth sensor.



(a) Initial velocity (b) Shifted velocity

Fig. 5. The observed corridor is limited by a maximum distance from the robot, a maximum distance to the velocity vector, a maximum angle to the velocity vector and a radius around the target. All laser scan points within this corridor are regarded as obstacles and have to be avoided.

To avoid obstacles a corridor surrounding the velocity vector is observed. If laser scan points are detected within this corridor the velocity vector is shifted until the corridor is free or until an abortion criteria is met (see figure 5 (b)).

4 Results

To determine the applicability and robustness of the projected interface in combination with the user input detection the user interface was projected onto a plane as described in section 2.2. The interface was divided into six areas which had to be selected by the user to generate different commands. Overall $n = 1080$ user inputs given by hand were evaluated and used to determine the influence of the relative position between the projection system and the projection plane as shown in figure 6.

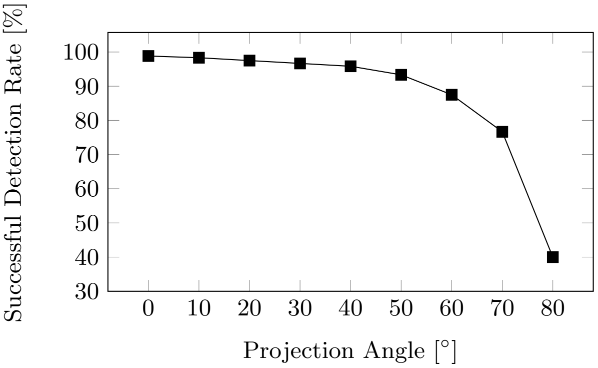


Fig. 6. Successful User Input Detection Rate Depending on the Projection Angle

As it can be seen, the average successful detection rate drops significantly if the projection angle becomes too steep and exceeds 50 degrees. Since the input was generated using hands, part of the increasing error may result from the movement of the hands over the desired interface area before actually touching it. The likelihood of generating a false input this way increases with the projection angle since the space above certain interface areas may occlude other areas. On the other hand the very high rate of successful detections remains relatively constant up to an angle of 30 degrees which is well suitable for most application cases.

5 Conclusion

In our approach we implemented an innovative interface for human-machine interaction. Using a camera-projector-system a graphical user interface is projected onto a suitable surface. The detected input enables the user to control a robot without the requirement of a special input device. The addition of the camera-projector-system to the robots manipulator allows for a high flexibility in the determination of a suitable projection area. In combination with our human tracking algorithm it allows for intuitive control of the robot in various environments.

The developed components are able to operate independently from each other allowing for transfer to any other mobile or stationary robotic system.

Increased robustness, though, especially concerning the user input detection may be achieved by making use of more advanced equipment and computing resources. Possible enhancements of the proposed system include the usage of a projecting device with increased brightness and the replacement of the 2D camera of the camera-projector-system with an RGB-D camera to further improve the determination of plane projection surfaces as well as the precision of the user input detection.

References

1. Lenz, C., Nair, S., Rickert, M., Knoll, A., Rösel, W., Gast, J., Bannat, A., Wallhoff, F.: Joint-Action for Humans and Industrial Robots for Assembly Tasks. In: The 17th IEEE International Conference on Robot and Human Interactive Communication, Munich (2008)
2. De Luca, A., Flacco, F.: Integrated control for pHRI: Collision avoidance, detection, reaction and collaboration. In: The 4th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob), Rome (2012)
3. Matsumaru, K.: Mobile Robot with Preliminary-announcement and Display Function of Forthcoming Motion using Projection Equipment. In: The 15th IEEE International Symposium on Robot and Human Interactive Communication, pp. 442–450. IEEE Press, New York (2006)
4. Park, J., Kim, G.J.: Robots with Projectors: An Alternative to Anthropomorphic HRI. In: Proceedings of the 4th ACM/IEEE International Conference on Human Robot Interaction, pp. 221–222. ACM, New York (2009)
5. Hosoi, K., Dao, V.N., Mori, A., Sugimoto, M.: VisiCon: A Robot Control Interface for Visualizing Manipulation Using a Handheld Projector. In: Proceedings of the International Conference on Advances in Computer Entertainment Technology, pp. 99–106. ACM, New York (2007)
6. Raskar, R., Beardsley, P.A.: A self-correcting projector. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 504–508. IEEE Press, New York (2001)
7. Leung, M.C., Lee, K.-K., Wong, K.H., Chang, M.-Y.: A Projector-based Movable Hand-held Display System. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1109–1114. IEEE Press, New York (2009)
8. Zhang, Z.: A Flexible New Technique for Camera Calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 1330–1334 (2000)
9. Harris, C., Stephens, M.: A Combined Corner and Edge Detector. In: Proceedings of the 4th Alvey Vision Conference, Manchester, pp. 147–151 (1988)
10. Duda, R.O., Hart, P.E.: Use of the Hough Transformation to Detect Lines and Curves in Pictures. Communications of the ACM 15, 11–15 (1972)
11. Zivkovic, Z.: Improved Adaptive Gaussian Mixture Model for Background Subtraction. In: Proceedings of the 17th International Conference on Pattern Recognition, vol. 2, pp. 28–31. IEEE Press, New York (2004)