Tracking body motions in order to guide a robot using the time of flight technology

Feryel Zoghlami[1], Harald Heinrich[2], Germar Schneider[3]and Mohamed Ali Hamdi[4]

[1,2,3] Infineon Technologies Dresden GmbH & Co KG, Königsbrücker Str. 180, 01099 Dresden, Germany
[1,2,3] (feriel.zoghlami,harald.heinrich,germar.schneider)@infineon.com
[2] National Institute of Applied Sciences and Technology
676 Centre Urbain Nord BP Tunis, Tunisia
mohamedalihamdi@yahoo.fr

Abstract. Nowadays, in many industries, humans share the working areas with intelligent robots within the manufacturing bays to ensure a collaborative effective work. However, programming robotic systems remains a challenging topic due to its costs and the high complexity of the programming resources needed for each operational task. In this context, the gesture would present a powerful interface tool to support the communication between humans and robots. We present in the first part a depth Time of Flight (ToF) based camera solution using machine learning for human body parts recognition and localization in the 3-dimensional space. In the second part, we use this information to teach robot movements and control its behavior in real time. Our solution recognizes with an accuracy of 90% the hand. It locates the human body parts and transfers continuously and in real time the actual location in order to guide the robot movements. In addition, our solution is also dedicated to proceed and classify two hand gestures used to monitor a handling robotic arm. The application supports the engineers to reduce the development time.

Keywords: First Keyword, Second Keyword, Third Keyword.

1 Introduction

Hand gestures are a natural form of human communication. For fabs where working in a shared area is needed, one person can use his body to interact with the machine. Therefore, gesture represents here a solid interface tool to convert this gesture into a relevant action. Computer vision has been explored in this topic and investigations were orientated towards solving the lack of interaction between humans and robots. The choice of using a Time of Flight (ToF) camera in our current application was based mainly on the advanced features offered by this technology. The camera provides the possibility of constructing a point cloud surrounding an object. As a result, direct processing of the distance information is possible with exact 3D location of each pixel with a measurement accuracy in the range of ±5mm. The camera used in this work presents

also a robust behavior regarding the illumination changing conditions for indoor applications and allows a possible parameters configuration monitoring.

The organization of the paper is as follow. In section II, we present a literature survey about related applications and methods used during the recent years. The hardware and the algorithms implemented in every phase of the developed solution are illustrated in section III. This is followed by the different experiments and their corresponding results described with brief discussions in section IV. We close up with a conclusion and outline the further future research trends.

2 Related Work

An effective collaboration between working humans and industrial robots needs frequent interactions between both parts. In this context, researches were focusing in the recent years on developing methods to transfer human motions to robots. The procedure starts with locating the human body in the space. Cristian Cnton-Ferrer [1] used to place colored markers in different positions on the human body in order to facilitate the body parts tracking. Adrian B. introduced in his article [2] a convolutional neural network (CNN) cascaded architecture specifically designed for learning part relationships and refining poses. However, the architecture of a neural network can be complex, whereas it is possible to track in the space the human body parts only with reference to the information received from a depth camera. In previous studies, numerous researchers have adopted joysticks and haptic devices to teach and control the robots. In [3] the authors explain a framework for robots manipulation using a joystick. This playback mode is particularly suitable for a humanoid robot, and the utility model has the advantage of long distance control like for games, which is not a common scenario in industries. However, for applications that do not require very high precision in industries, speech recognition like explained in [4] becomes a way to teach robots based on natural perception channels of human beings. This approach is intuitive, however in industrial environment it can be problematic because of loud noises and low precision of voice commands. Guanglong Du [5] combines speech and gesture for online teaching. He used a Kinect depth camera and an inertial measurement unit to capture the human-natural interactions and transfer it in real time to robots. Other researchers chose to use wearables (Jackets and gloves) to teach robots. However, they had difficulties in installing sensors in fixed positions inside the wearables, which lower the precision of the system.

For gesture recognition, several techniques have been adapted for years. It started with using colorful gloves for hand and fingers tracking [6]. This method suffers from instability in the system. In 2007, Breuer P introduced a new framework using a ToF camera in recognizing the hand shape [7] and the results proved less complexity and more accuracy. Several other authors have emphasized the importance of using many diverse training examples for 2D and 3D CNNs [8, 9]. The results shows high accuracy for hand gesture recognition tasks with usage of both RGB and depth cameras. How-

ever, neural networks have difficulties in dealing with objects of various scales compared to other algorithms like Haar, histogram of oriented gradients (HOG) and local binary pattern (LBP) introduced in [10]. Neural networks are also mainly used in classification tasks but it demands a large dataset. However, with a limited dataset the reference to the support vector machine SVM algorithm would be a better choice for high precision and overfitting scenario avoidance. George E. Sakr Compares in his paper [11] between both algorithms' performances for waste sorting problem.

3 Methodology

We describe the methods to define the algorithmic components of our human body parts recognition and localization solution.

3.1 Hardware Choice

Our proposed solution is based on the usage of a PMD CamBoard pico flex ToF camera, with a resolution 224 x 171 pixels. We chose for our application to work with a frame rate equal to 35fps. It shows a 100 times higher depth accuracy than a stereo camera and finally it is less sensitive to mechanical alignment and environmental lighting conditions for indoor applications. The dimension of the camera is 62 mm x 66 mm x 29 mm, which make the implementation on a moving robotic arm easy. We run all the training algorithms and the tests on a CPU Intel core i5.

3.2 Human Body Parts Tracking

The usage of the geodesic distance is a way to construct a graph representation of the 3D points that is invariant to articulation changes and, thus, allows identifying the human body parts regardless his pose or his position. The geodesic distance is defined as the shortest path between two points p and q on a curved surface and according to Loren A.S. Artashes M. in their paper [12], it is calculated as follows:

$$d_G = \sum_{e \epsilon SP(p,q)} \omega(e) \quad (1)$$

Where e represents the edge that links p and q, $\omega(e)$ is the possible Euclidian distances between the two points going through different edges and $SP(p,q)$ contains all edges along the shortest path between p and q. Considering two separate 3D points $p(p_1, p_2, p_3)$ and $q(q_1, q_2, q_3)$ in the 3D coordinate system of the camera, the computation of the Euclidian distance is given by the Pythagorean formula:

$$d(p,q)=d(q,p)= \sqrt{\sum_{i=1}^{n} (p_i-q_i)^2} \quad (2)$$

We refer to the Dijkstra's algorithm to construct the geodesic graph from the depth image retrieved from the ToF camera. As an output, the algorithm helps in locating the different parts of the human body in the space including the hands. These parts are simply tracked on the geodesic graph. The considered algorithm presents a low

computational time ~230ms when we consider a limited number of 3D points in the depth image. Before we construct the geodesic graph, we construct the vertices-edges graph $G_t = (V_t, E_t)$, where $V_t=\{(x_i,y_j)\}_t$ is the vector of the considered points in the 2D image, which we call vertices and $E_t \subseteq V_t x V_t$ is the group of the edges, which present the Euclidian distance between two vertices. E_t is determined as follows [12]

$$E_t=\{((x_i,y_j),(x_k,y_l)) \epsilon V_t x V_t \quad d((x_i,y_j),(x_k,y_l))<\delta \wedge \|(i,j)^T-(k,l)^T\|_\infty <1\} \tag{3}$$

$\|.\|_\infty$ is the maximum norm and $(i,j)^T$,$(k,l)^T$ are the 2D coordinates of the two points (x_i, y_j) and (x_k, y_l) in the depth image. δ is a threshold up to it a distance is considered edge.

The Dijkstra's algorithm in this case is summarized in the flow chart Fig.1.

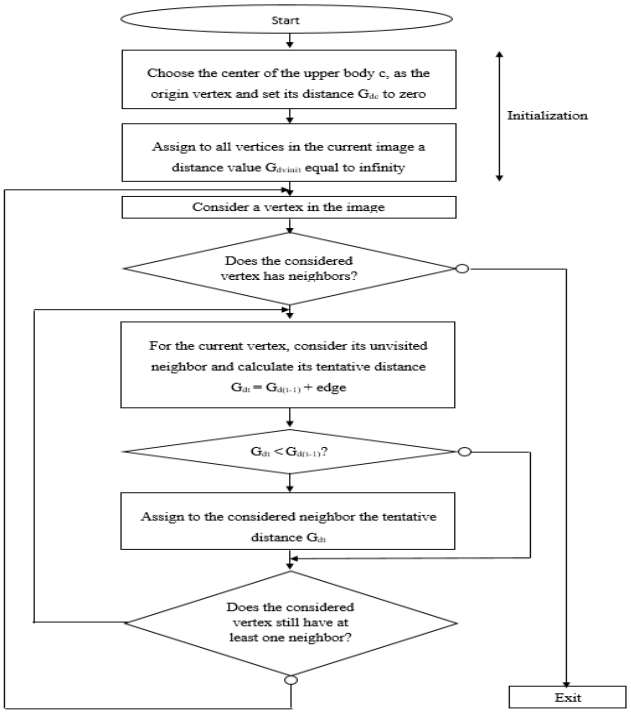

Fig. 1. Flowchart of the geodesic graph construction

3.3 Hand Shape Recognition

In our solution, we build for the human part recognition (in our case the hand) both Haar and Local Binary Pattern (LBP) feature-based cascade classifiers. Therefore, we collect 2D amplitude (Greyscale) images from the camera and we organize them into two categories: cropped positive images that present the hand in different scales and different orientations and negative images, which include objects others than a hand. We feed the image dataset into one of the two algorithms mentioned above that extracts

relevant features from the available images based on the variation of intensity in different regions of pixels and save these feature values in an xml file.

Preprocessing Before moving to the feature extraction phase, we work on the noise reduction. Due to the presence of reflective objects in the environment, the images received from the ToF camera can present noisy regions. To reduce the noise, two different techniques are applied for every received frame:

- The application of a median filter, which helps in removing the salt and pepper noise from the amplitude.
- The ToF cameras have the advantage of allowing the user to set the desired exposure time. By setting a low exposure time (200ms), we can reduce the noise in the image.

We move next to the background to make the processing simpler. For the proposed solution, we focus our interest on a distance up to 1 meter in front of the camera.

3.4 Hand Localization

Once recognized in real time we draw a bounding box around the hand and we consider this box as our new Region Of Interest (ROI). We develop an algorithm that structures in a two-dimensional array the pixels that define the contour of the largest object in the ROI, which would be certainly our recognized hand. We determine later the center of mass of the hand (the hand is considered as a rigid corps and we define it by its center of mass). As a result, we get the x_c, y_c and z_c coordinates of the center of mass of the hand from the depth confidence image retrieved from the ToF camera. For a better localization in the space, we consider another point $p(x_p, y_p, z_p)$ on the hand and we determine the orientation of the hand as follows:

$$\begin{cases} \alpha = Cos^{-1}\left(\frac{y_p - y_c}{z_p - z_c}\right) \\ \beta = Cos^{-1}\left(\frac{x_p - x_c}{z_p - z_c}\right) \\ \gamma = Cos^{-1}\left(\frac{y_p - y_c}{x_p - x_c}\right) \end{cases} \tag{4}$$

3.5 Gesture Classification

We limited our classification to two classes: palm and fist starting by capturing 100 training greyscale images for each hand posture with different lighting. The considered training stages are illustrated in Fig.2.

The algorithm starts with the extraction of SIFT features for each training image following four successive stages:

- Find the locations of potential interesting points (key points) in the image.
- Refine the key points' locations by eliminating points of low contrast.
- Assign an orientation to each key point based on local image features.
- Compute a local feature descriptor at each key point.

In the next step, we cluster our extracted features with k-mean clustering algorithm:

- Choose randomly 4 centroids, which will present the centers of each cluster.

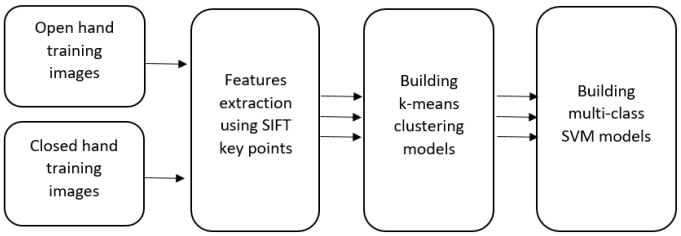

Fig. 2. Algorithm stages for gesture classification

- Assign every key point (defined in the previous step) to the nearest centroid.
- Shift the centroids to the average location of all the key points assigned to them and repeating this procedure until the assignments stop changing.

In the end for each training image we get 4 clusters and each cluster is defined by a vector whose elements are the extracted key points from the previous step. Afterwards, depending on the hand posture, images of the same category are then labeled with the same number: Label or class 1 for the fist training images, class 2 for the palm training images. The final step consists in using the hand posture images with their assigned labels to train a multiclass SVM model. We chose to set the penalty factor C value to 10^3 in order to optimize our model in terms of lowering misclassifications within the training data. We defined that our training run for maximum 100 iterations and with an error tolerance ε of 10^{-6}.

3.6 Human Upper Body Tracking

Once constructed we detect on the geodesic graph the body parts of the human who is standing in an idle pose by looking for the largest geodesic distances from the center in all directions: up, left, right, down left, down right. They correspond respectively to the location of the head, the left hand, the right hand, the left foot, the right foot The limited field of view of the used pico flexx ToF camera (62° x 45°) allows a restricted human upper body tracking.

3.7 Hand Shape Recognition

We prepare for testing 4 classifiers: two HAAR feature-based classifiers and two LBP feature-based classifiers trained each on a dataset composed of 1200 positive samples and 1000 negative samples. The evaluation of the 4 classifiers is based on a test done first on 200 and second on 500 successive frames. We look with the camera in areas including different objects (hands and other objects).

The performance of the cascade classifier depends on the feature-based algorithm used during the training and the number of stages adopted to create the classifier. Over-training can create a weak classifier. It is the case of the LBP classifier trained for 20 stages. Based on the previous results, we calculate different performance metrics presented in the table 1.

The fourth LBP feature-based classifier shows the best performance in a range between 100mm and 600mm in front of the camera with the highest recognition accuracy ~90% and a low false positive rate <0,02. Using this classifier, the recognition rate is equal to 26ms.

Table 1. Performances of pre-trained cascade classifiers

Classifier	NbFrames	TP	TN	FP	FN	Recall	Precision	Accuracy	TPR	Average Processing time (millisecond)
1	200	184	168	32	16	0.92	0.852	0.88	0.92	27
	500	412	459	41	88	0.824	0.887	0.871	0.824	27
2	200	159	176	22	41	0.795	0.878	0.838	0.795	26
	500	422	472	28	78	0.844	0.938	0.894	0.844	26
3	200	161	187	14	39	0.805	0.92	0.87	0.805	27
	500	434	450	50	66	0.868	0.897	0.884	0.868	27
4	200	163	197	3	37	0.815	0.982	0.9	0.815	30
	500	343	475	25	158	0.684	0.932	0.81	0.684	30

TP:True Positive; TN:True Negative; FP:False Positive; FN:False Negative; TPR:True Positive Rate

3.8 Gesture Classification

We evaluate the performance of the SVM model by testing 50 images from each hand posture. The results show 86% right classification for the open hand and 76% right classification for the closed hand. The classification process takes 18ms for every testing image and it will be tested to control handling tasks for an automated robotic arm.

3.9 Data Transmission

Once the hand has been recognized, classified and located in the space we construct a message with all information about the hand and serialize it in a binary form via protobuffer to a global server. We use the MQTT protocol for data transmission and we use for testing a Kuka robot system with a pico flexx installed on a gripper flange. The robot connects to the same server and gets in real time the message sent by the camera. Tests indicate that the robot follows the hand movements in all directions and the whole process takes 300 milliseconds. The test was executed by kind support at the laboratory of the company Wandelbots GmbH Dresden.

4 Conclusion And Next Steps

We present a very promising solution for human gesture recognition that can be used for industrial robots guidance. The system's design fuses the information received from a Time of Flight camera together with sophisticated machine learning techniques. Our evaluation proves that its robustness, accuracy, efficiency and cost-effective characteristics make it a suitable framework for applications in robotic perception and interaction for collaborative industrial robots. The developed system contributes as an alternative approach for teaching robots movements without referring to complex programming. Our solution is adaptive to different robots models and can help to teach or control robots without additional teaching tools (joystick, trackball...).

To deal with the limitations of our system regarding the small opening view of the camera and the limited detecting range, an interesting future direction will be to apply advanced machine learning algorithms in a compact sensor data fusion system. This

system will not only increase the perception rate and decrease uncertainties in a complex environment for industrial shared areas between robots and humans but will also help robots to acquire intelligence in order to establish an effective collaborative work.

Acknowledgment

The co-funded innovation project Arrowhead-Tools receives grants from the European Commissions H2020 research and innovation programme, ECSEL Joint Undertaking (project no. 826452), the free state of Saxony, the German Federal Ministry of Education and national funding authorities from involved countries.

References

1. Cristian Cnton-Ferrer, J. R., "Marker-Based Human Motion Capture in Multiview Sequences", EURASIP Journal on Advances in Signal Processing. Vol. 2010, pp. 11, article No. 73, 2010.
2. Adrian B., Georgios T. "Human Pose Estimation via Convolutional Part Heatmap Regression", Springer International Publishing. Vol. 9911, pp. 717-732, 2016.
3. K. B. Cho and B. H. Lee, "Intelligent lead: A novel HRI sensor for guide robots" Sensors, vol. 12, no. 6, pp. 8301–8318, 2012.
4. V. Gonzalez-Pacheco, M. Malfaz, F. Fernandez, and M. A. Salichs, "Teaching human poses interactively to a social robot," Sensors, vol. 13, no. 9, pp. 12406–12430, 2013.
5. Guanglong Du , Mingxuan Chen, Caibing Liu, Bo Zhang, and Ping Zhang, "Online Robot Teaching With Natural Human–Robot Interaction" IEEE transactions on Industrial electronics. Vol 65,pp. 9571 – 9581, no. 12, 2018.
6. Luigi Lamberti, Francesco Camastra ., "Real-Time Hand Gesture Recognition Using a Color Glove ". International Conference on Image Analysis and Processing ICIAP2011. Vol 6978, pp 365-373, 2011.
7. Breuer, P. a., "Hand Gesture Recognition with a novel IR Time-of-Flight Range Camera". International Conference on Computer Vision / Computer Graphics Collaboration Techniques and Applications MIRAGE 2007. Vol. 4418, pp 247-260 2007.
8. Ali A. Alani ; Georgina Cosma ; Aboozar Taherkhani "Hand gesture recognition using an adapted convolutional neural network with data augmentation", 4th International Conference on Information Management (ICIM). Pp 5-12, 2018.
9. Aparna Mohanty, "Deep Gesture: Static Hand Gesture Recognition Using CNN" Palmprint Recognition based on Minutiae Quadruplets. Vol 460, pp 449-461, 2017.
10. Mahmood Jasim, Tao Zhang and Md. Hasanuzzaman "A Real-Time Computer Vision-Based Static and Dynamic Hand Gesture Recognition System", International Journal of Image and Graphics. Vol. 14, no. 01n02 2014.
11. George E. Sakr , Maria Mokbel, Ahmad Darwich , "Comparing deep learning and support vector machines for autonomous waste sorting". IEEE International Multidisciplinary Conference on Engineering Technology (IMCET).Pp. 7, 2016
12. Loren A.S. Artashes M., "Estimating Human 3D Pose from Time-of-Flight Images Based on Geodesic Distances and Optical Flow". Ninth IEEE International Conference on Auto matic Face and Gesture Recognition. Pp. 7, 2016.