Recognizing Hand Gestures for Human-Robot Interaction

Zuhair Zafar and Karsten Berns

RRLAB, Department of Computer Science,
Technical University of Kaiserslautern,
Kaiserslautern, Germany
Email: {zafar,berns}@cs.uni-kl.de

***Abstract*—Human-Robot Interaction is the most important aspect for the development of social service robots. Interacting with social robots via non-verbal communication makes the interaction natural and efficient for human. We present an interface that uses hand gestures to interact with humanoid robot. The major goal of this interface is to recognize gestures in dynamic environments with high accuracy and efficiency. Our proposed system enables automatic recognition of 18 different human hand gestures from RGB-D (color and depth data) device. Robot expresses different facial expressions and performs the gestures after recognizing them. We use bag-of-features approach to recognize gestures using scale invariant feature transform (SIFT) keypoints. The system is invariant to scale, slight rotation and illumination and can work in cluttered backgrounds. We use multi-class support vector machines for classification task. In order to validate our scheme, we use this interface in our humanoid robot that reports more than 94% recognition rate for 18 hand gestures.**

Keywords–Human-Robot Interaction; Nonverbal communication; Hand Gestures; Bag-of-features approach.

I. Introduction

Since last decade, many different types of human-friendly robots have been developed. Varying in the objectives, some robots are developed for helping humans in industrial environment and some are designed to function in indoor environment. As the technology gets sophisticated and more advanced, the focus has been shifted to social service robots. The goal of these robots is to communicate with human in a human-like way and perform different tasks as instructed by human. This leads us to social behavior in robots. These social robots should recognize humans, their verbal communication and gestures in order to realize natural communication. Furthermore, they should also recognize human emotions in order to predict the internal state of human for better communication.

Social behaviour in robots generally depend upon efficient human-robot interaction (HRI). The most common way of human interaction is either by vocal communication or by body gestures. Other medium includes newspapers, notes and other writing material, however, this type of communication is not applicable in face-to-face communication. According to [1], 65% of our communication consists of human gestures and only 35% consists of verbal content. This two third of our communication shows the significance of gestures. For this purpose, recognition of nonverbal content becomes essential task for HRI. Human gestures are nonverbal content, which are used with or without verbal communication in expressing the intended meaning of the speech. Gestures may include hand, arm, or body gestures and it may also include eyes, face, head etc.

Human gesture recognition has been a popular topic in computer vision field. The topic has been studied numerous times because of its important applications in surveillance systems, elderly care, in the field of medicine (e.g. gait analysis, surgical navigation), in the field of sports, augmented reality, sign language for hearing impaired people and human behavior analysis. Hand gestures are critical in face-to-face communication scenarios. Especially during discussions, hand gestures become more animated. They emphasize points and convey enthusiasm of the speaker. Hand gestures show a lot about the internal state. For example, crossing arms during face-to-face communication shows nervousness or lack of interest and clenched hands show aggressive stance of a person. They enable human to express mood state (like thumbs up) or convey some basic cardinal information (like one, two and so on). In this paper, we present a system, which enables a humanoid robot to recognize different hand gestures used in daily routine. Not only the robot recognizes them, but it expresses its emotions through its facial expressions accordingly. We also have conducted several experiments where robot is able to imitate the same gesture in real time.

This paper is organized as follows: Section 2 discusses the existing hand gesture recognition systems in context to HRI. We describe our gesture recognition approach in Section 3 and evaluate our methodology in Section 4. We conclude our paper in Section 5.

II. Related Work

Numerous hand gesture recognition systems have been reported in the literature. In general, we can categorize them in two different classes: (a) data gloves based systems and (b) vision based systems. Former type of systems require use of glove sensor for storing hand and finger motion and then use this data to recognize the action. Huang et al. [2] used gloves to record the hand and fingers flex data and then use machine learning algorithms to classify 5 dimensional finger flex data. Although, this type of systems may provide a 3D representation of hand however, wearing a heavy and expensive glove is not suitable for natural human interaction. On the other side, vision based systems take the information of the hand itself as an input using a camera to collect hand movements for gesture recognition without the use of any wearable sensor. Vision based approaches can be divided into two categories, i.e., 3D hand model based method and appearance based methods. The 3D hand model can provide ample information of hand that allows to realize wide class of hand gestures but the main disadvantage lies in extraction of features in case of ambiguous poses, unclear views and high computational complexity, which makes the overall system unrealistic for real time interaction.

In appearance based approaches, images with hands are considered only for feature extraction and gesture recognition task. The simplest technique is to look for skin color regions. Marilly et al. [3] extracts hand region using skin color and foreground information. For feature extraction, they use statistical and geometric features and then classify the gestures using principle component analysis. However, this method has some serious shortcomings. The major drawback of color-based techniques is the variability of the skin color in different lighting conditions. This frequently results in undetected skin regions or falsely detected non-skin textures. The problem can be somewhat alleviated by considering only the regions of a certain size (scale filtering) or at certain spatial position (positional filtering).

Another appearance based approach presented in the literature [4], that uses Gabor filters for extraction of hand gesture features. Gabor filters can capture the most important visual properties such as spatial locality, orientation selectivity and spatial frequency. Due to the high dimensionality of features, principle component analysis is used for feature reduction. The drawback of this and other similar approaches is that these methods are not invariant to translation, rotation and scaling. Moreover, these approaches are also effected by illumination variation. In [5], cascade of classifier approach is used. Each cascade is capable of detecting hands with certain angle of rotation. The drawback of this approach is that, it can not classify the same gesture with different viewpoints and is not rotation invariant. The authors of [6] extract a distinct and unified hand contour to recognize hand gestures, and then compute the curvature of each point on the contour. Due to noise and unstable lighting in the cluttered background, it has difficulties in obtaining segmentation of integrated hand contour. The eigen space is another technique, which presents a robust representation of a huge feature set of high-dimensional points using a small set of basis vectors. However, eigen space methods are not invariant to translation, scaling, and rotation. The most common and serious shortcoming of all of these methods discussed so far is that they only work with uniform background. These approaches lack in detecting hands in cluttered and dynamic environment.

Local invariant features are used for object recognition task. In the paper [7], to perform reliable matching between different views of an object or scene, a method is presented for extracting distinctive invariant features, as known as scale invariant feature transform (SIFT) features, that can be used for object recognition. This method for image feature extraction transforms an image into a large collection of feature vectors, each of which is invariant to image translation, scaling, and rotation, partially invariant to illumination changes and robust to local geometric distortion. Hartanto et al. [8] use SIFT features along with skin detection method for background subtraction and contours for localization of hand. Their matching stage is relatively simpler and hence, reports less than 70% accuracy for Indonesian sign language database. Their approach is computationally extensive and is not applicable for real time recognition. Dardas and Georganas [9] used bag-of-features approach using SIFT features as keypoints and then used support vector machines to recognize the hand gesture. They segment the hand based on the skin color, discard the face using Viola-Jones face detector and then extract features. The shortcoming of this approach lies in segmentation of hand. It depends highly on illumination variation and the subject wearing half or full sleeves.

In order to address all of these shortcomings, we proposed to use 3D camera (e.g., ASUS Xtion Pro) to localize hands using joint information. We use depth data to locate hands using OpenNI and NiTe middleware library and their positions for hand segmentation. We describe our approach in detail in the following section.

III. HUMAN HAND GESTURE RECOGNITION

Visual perception in complex and dynamical scenes with cluttered backgrounds is a very difficult task, which humans can solve satisfactorily. However, for a robot perception system, it performs poorly in this kind of challenging scenarios. One of the reasons of this large difference in performance is the use of context or contextual information by humans. Several studies in human perception have shown that the human visual system makes extensive use of the strong relationships between objects and their environment for facilitating the object detection and perception. Not only this, due to the notion of real time, robot has to perform its computations as fast as possible. Due to which, most of the time robot perception system is hampered with low resolution images. There is a need to develop such perception system, which can cater complex environments and work efficiently. Keeping this in mind, we propose an approach that uses depth sensor for hand segmentation using NiTe library. It uses bag-of-features approach from SIFT keypoints and classify them using support vector machines (SVMs) to recognize hand gestures. The block diagram of the approach is presented in Figure 1.

Our proposed system is highly robust and efficient. It reports 94% recognition rate for 18 different gestures when classified by multi-class support vector machines. Hand gestures are robustly recognized with dynamic and cluttered background. The system is invariant to illumination, scale and slight rotation. Figure 1 shows the schematic flow of the system. Each module of the system is described in the following sub-sections.

A. Pre-processing

Instead of using monocular camera, ASUS Xtion is used in order to use depth data. The advantage of using such devices with depth sensor lies in localization of hands. Segmenting hand on the basis of skin color can be useful in a constraint scenario but it behaves poorly when applied in dynamic environment. While using depth sensor, hands can be localized efficiently in cluttered and dynamic environments. RGB-D device delivers color and depth stream of the scene. Before extracting features directly for further processing, preprocessing is an essential step in registering images. We propose to use lower resolution 320×240 in order to reduce the computational complexity of the system.

B. Hand Detection and Segmentation

As discussed earlier, NiTe middleware library enables us to detect hands by utilizing depth and IR sensor of ASUS Xtion. Not only the algorithm detects hands but also tracks them efficiently. In our setup, ASUS Xtion sensor is used, which delivers the centroid of the hands using depth data. The

 ISBN: 978-1-61208-468-8

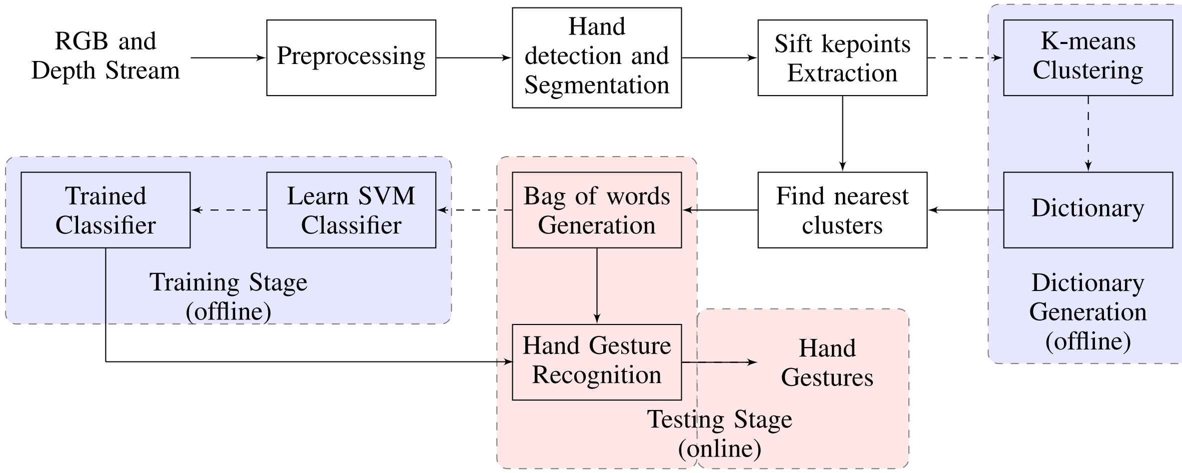
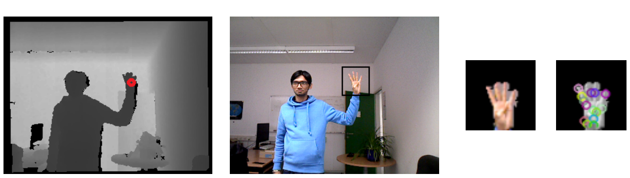


Figure 1. Working schematics of our approach. Using depth stream, hand is localized and segmented, SIFT keypoints are extracted. A dictionary is generated using k-means clustering; bag of words vectors are generated using cluster model; which are fed into SVMs for classification

algorithm can detect multiple hands if present in the scene. The most nearest hand on the basis of the depth data from the robot is selected. In order to segment the hand out of the whole image frame, firstly color and depth streams are synchronized and then a small region around the centroid of the hand is extracted from color image. This small region depends on the depth data. If a person is near the robot, this region of interest becomes bigger and if a person is far from the robot, this region of interest is small. The extracted region of interest has some irrelevant information other than the hand. In order to discard that information, pixels in the region of interest which have depth close to hand's depth are kept and everything else is discarded. Figure 2 shows the original image and hand segmented image. In order to make it scale invariant, each segmented image is registered by resizing it to 50×50 resolution.

C. Bag-of-Features Extraction

Bag-of-features (BoF) is probably the most popular technique of feature representation for videos and still images in the domain of human gesture recognition. Coming from the text mining paradigm, BoF representations allow to recognize a variety of gestures ranging from simple periodic motions (walking, running) to interactions (waiving, shaking hands). It is usually combined with other procedures in feature extraction task. The schematics include (a) keypoints extraction, (b) descriptor generation around those point, (c) dictionary generation and (d) use of clustering algorithm to create BoF vectors. In order to represent an image in BoF approach, it can be expressed as a document. Consequently, the features extracted can be expressed as words in this document. All the words or features extracted from the document or image if combined together, form a dictionary.



(a) (b) (c) (d)

Figure 2. (a) Hand is detected in depth stream, (b) Hand is localized in color stream based on its position (c) segmented hand after discarding background with higher depth than hand's depth and (d) SIFT keypoints extracted

1) Keypoints Extraction: The initial step in BoF algorithm is to extract keypoints. These keypoints should be able to represent the whole image. For extraction of keypoints, SIFT algorithm is used. SIFT has been quite popular in object recognition and scene classification because of its robustness and efficiency. Proposed by Lowe [10], SIFT algorithm selects the features that are invariant to image scaling and rotation and to some extent illumination changes. An image is convoluted with Gaussian channels at various scales. Keypoints are taken as the maxima or minima points of the difference of Gaussian that happen at various scales. A lot of keypoints are discarded, which are not stable or have low contrast using Taylor series expansion. Figure 2 shows the extracted keypoints on a segmented image.

2) Descriptor Generation: The keypoint descriptor is computed by calculating gradient magnitude and orientation of all the keypoints around its location. The orientation histograms are relative to the keypoint location. The histogram contains 8 bins and computes array of 4×4 histogram around the keypoint. Since there are 4×4 histograms each with 8 bins, the keypoint vector is 128 dimensional. This vector is then normalized to unit length in order to boost invariance to changes in illumination. Hence, the features are illumination invariant.

3) Dictionary Generation: This is an important step in the BoF algortihm. The size of the dictionary is critical for the recognition process. If the size of the dictionary is set too small then the BoF model can not express all the keypoints and if it is set too high then it might lead to over-fitting and increasing the complexity of the system. K-means clustering algorithm is used to cluster keypoint descriptors of all the training images in k clusters, where k is the dictionary size. The centroids

Copyright (c) IARIA, 2016. ISBN: 978-1-61208-468-8

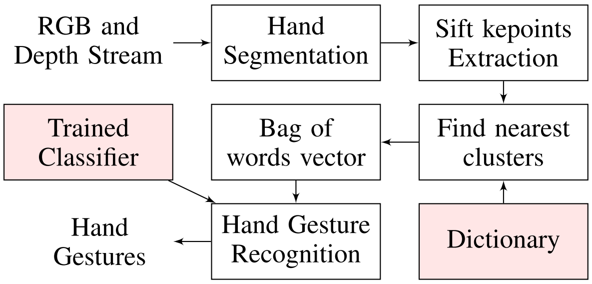


Figure 3. Block diagram of testing stage

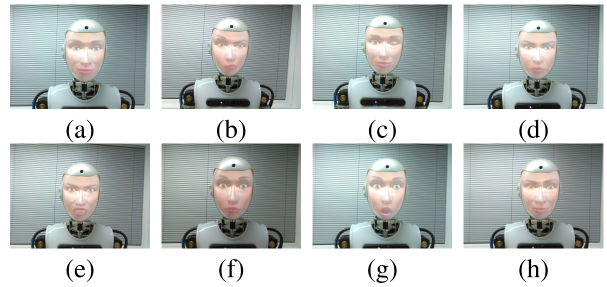


Figure 4. Robot performing facial expressions (a) Happy (b) Thinking (c) Wink (d) Frown (e) Threaten (f) Surprise (g) Astonishment (h) Disgust.

of each clusters are combined together to make a dictionary. This resulted in dictionary with $k \times 128$ size. Dictionary is generated offline and is used for making bag of words vector. In our study, $k = 400$ is used as a dictionary size.

4) Bag of Words Vector: In order to make bag of word vector, SIFT keypoints are extracted for hand segmented image. Each keypoint descriptor of the image is compared with each centroid of the cluster in the dictionary using euclidean measure. If the difference is small or keypoint is closed to a certain cluster, the count of that index is increased. Similarly other keypoints of an image are also compared and the counts of the respective indices are increased to which the keypoints are closest to. Finally, bag of words vector is generated for a single image that has size $1 \times k$ where k is the dictionary size. In our study, the size of bag of words vector becomes 1×400. One important thing to note here is the size of the feature vector. Using only SIFT keypoints, the size of feature vector is $n \times 128$ (where n is the number of keypoints) which is complex and makes the system computationally more intensive. On the other hand, BoF approach reports 400 dimensional feature vector which has the characteristics of SIFT features and considerably less complex. These bag of words vectors are computed for all the image frames for training and testing of the gestures.

D. Classification

Classification is an important step in any recognition task. There are a lot of classifiers presented and used according to the type of problem. We proposed to use SVM because of its effectiveness in high dimensional spaces [11]. SVM only uses small set of support vectors for decision making, hence it requires less memory and is used in real-time recognition tasks. With different kernel functions available, SVM can used for versatile problems. One-versus-all approach of multi-class SVM is used. Bag of words vectors for all the images are computed in training stage and labels are appended according to the class. This bag of words vectors are fed into the multi-class SVM in order to train the model that is further used in testing stage for hand gesture recognition. To validate our system, a database is generated. 600 images for each gesture are used for training. Each hand segmented image is manually selected and noisy or ambiguous images are discarded. Six different subjects have featured in the database. Each gesture is recorded at various scales and at different positions.

Figure 3 shows the working schematics of our testing phase. Color and depth frames are extracted from ASUS Xtion. Hand is localized and segmented in the same way during training stage. These segmented images are then used to extract keypoints using SIFT algorithm. The keypoints are compared with the predefined dictionary (created offline) and bag of word vectors are generated. For each testing image, the 400 dimensional feature vector is fed to SVM with already learned classifier. Hand gestures are then recognized in real-time. 18 different hand gestures have been recognized: Palm, pointing upwards, inquiring gesture, number gestures like one, two, three, four, five, thumbs up, fist, little finger, victory, etc.

IV. Experimentation and Evaluation

The goal of the system is to recognize hand gestures reliably and robustly in order to realize natural human-robot interaction. We use our humanoid robot in order to evaluate accuracy and efficiency of our system. University of Kaiserslautern is developing a social humanoid robot, Robothespian [12]. It consists of intelligent hands and arms and a backlit projected face. The whole arm has 14 degrees of freedom, where hands are able to perform nearly all type of hand gestures. The backlit projected head is able to express any facial expression using action units. Robot can also move torso and head to about ± 20 and ± 45 degrees respectively. ASUS Xtion is mounted on the chest of the robot. It has its own processor that can handle the movements of all the joints and facial expressions. It is used to evaluate our hand gesture system. In the following section, we describe the set of experiments conducted and then evaluate the performance of the system.

A. Experimental Setup

Two different experiments are conducted in our study to evaluate the hand gesture recognition system. 18 different hand gestures are used to interact with the humanoid robot. We divide them in two different categories. In the first experiment, common hand gestures that humans use during daily communication are used namely: thumbs up, fist, palm, inquire gesture, okay gesture, pointing gesture and little finger gesture. These gestures play an important role to express the inner thoughts or intentions of a human. For example, thumbs up gesture shows an approval, while fist gesture shows aggressiveness of human. In order to make HRI more natural, robot expresses facial expressions on its face according to the gesture. Table I shows different gestures and their corresponding expressions that are expressed by the robot and Figure 4 shows robot's facial expressions.

In the second experiment, basic gestures, that are used to express the fundamental information like numbers, e.g. one,

 ISBN: 978-1-61208-468-8

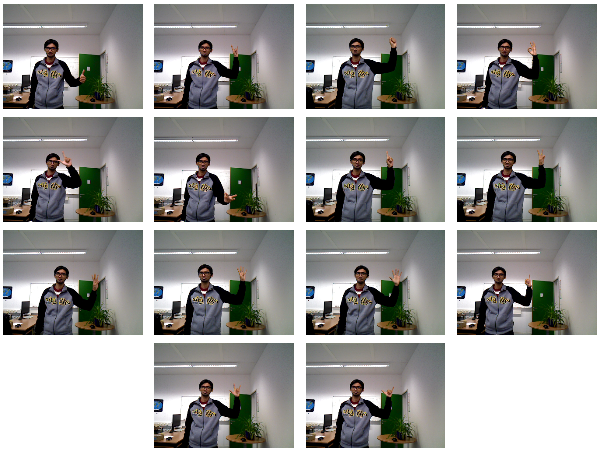


Figure 5. Subject performing different gestures in front of the robot.

two, three, four and so on, are recognized. These gestures are used to represent the numbers from one to ten using only single hand. Instead of expressing facial expressions, robot imitates the same *number* gesture as performed by human. Not only robot imitates the gesture but also verbally communicates the *number* using its built-in speaker. For natural interaction, the robot's head also focuses directly towards the hand position. In this way, wherever the hand is in the frame, robot's head move to that position as if to focus the hand. Figure 5 shows different hand gestures while Figure 7(c) shows robot's head is focusing the hand of the human in the frame.

TABLE I. DIFFERENT GESTURES AND CORRESPONDING FACE EXPRESSIONS EXPRESS BY THE ROBOT

Nr.	*Hand Gesture*	*Facial Expression*
1)	Okay gesture	Happy
2)	Pointing Up	Thinking
3)	Thumbs Up	Wink
4)	Inquiry gesture	Frown
5)	Fist	Threaten
6)	Palm	Surprise
7)	Victory	Astonishment
8)	Little finger gesture	Disgust

B. Performance Measurement

As discussed in the earlier section, the prime objective of the system is to recognize hand gestures in dynamic scenarios in context to HRI. System should recognize gestures in real time. Two set of experiments, discussed in previous section, are evaluated in order to measure the performance. For performance evaluation, the predicted gesture along with subsequent image of all the frames are stored in a separate file. In the end, we manually evaluate the results. In the first experiment, eight different gestures: thumbs up, okay gesture, pointing upwards, inquire gesture, victory, fist, palm and little finger gesture are recognized. A total of 1200 frames are stored during first experiment. Table II shows the recognition rates for each hand gesture.

As shown in the Table II, the average recognition rate in case of common hand gestures when performed in front of the robot is 93%. For Okay gesture, the recognition rate is low due to the reason that the fingers are pointing upwards

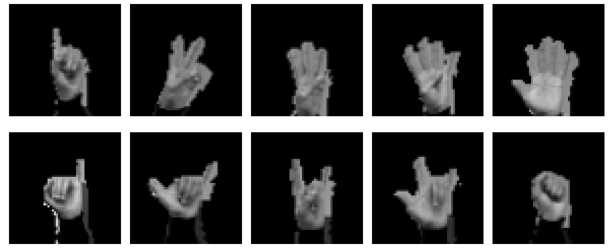


Figure 6. Hand segmented images of number gestures (1-10).

and sometimes it is confused with victory and inquire gesture. Gestures like thumbs up and little finger are occasionally confused with pointing up gesture. In all three cases, one finger is pointing upwards which results in false detection. All of the gestures are tested, when the gesture is performed dynamically or statically.

TABLE II. RECOGNITION RATES FOR COMMON HAND GESTURES.

Hand Gesture	*Number of Images*	*Correctly Classified*	Recog. Rate
Okay gesture	150	115	76.6%
Pointing Up	150	149	99.3%
Thumbs Up	150	136	90.7%
Inquiry gesture	150	141	94%
Fist	150	144	96%
Palm	150	150	100%
Victory	150	145	96.7%
Little finger gesture	150	137	91.3%
Average			**93%**

In the second experiment, *number* gestures are used for recognition task. Ten gestures are performed from a single hand to recognize numbers. Figure 6 shows ten segmented hand gestures. For this experiment, 1500 images are used to evaluate these gestures. Table III shows the number of correctly classified images along with the recognition rate of each gesture.

From Table III, it can be seen that *three* gesture has low recognition rate. According to experiments, this gesture is falsely detected as *four* gesture and occasionally, as *two* gesture. From the validation studies, we find out that performing *three* gesture perfectly is difficult for few subjects, which makes it harder to recognize. We also observe that *six* or little finger gesture is occasionally confused with *one* gesture. In both these gestures only one finger, index finger in case of *one* gesture and little finger in case of *six* gesture, is pointing up which easily confuses the gesture. The overall average recognition rate is 95.1%. Figure 7 shows our experimental

TABLE III. RECOGNITION RATES FOR NUMBER GESTURES.

Number Gesture	*Number of Images*	*Correctly Classified*	Recog. Rate
One	150	149	99.3%
Two	150	145	96.7%
Three	150	133	88.6%
Four gesture	150	134	89.3%
Five	150	150	100%
Six	150	137	91.33%
Seven	150	137	91.33%
Eight	150	148	98.7%
Nine	150	150	100%
Ten	150	144	96%
Average			**95.13%**

 ISBN: 978-1-61208-468-8

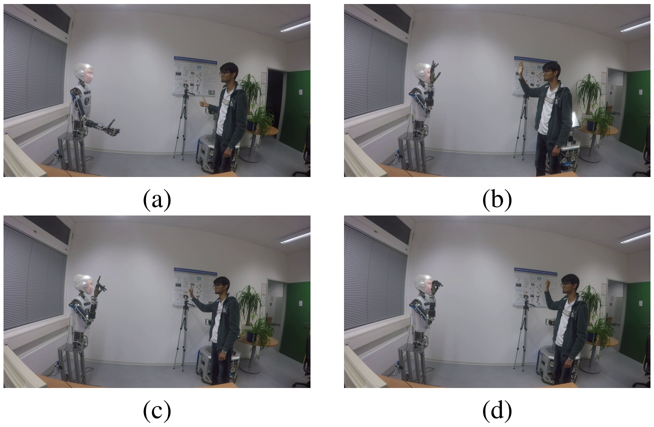


Figure 7. Robot recognizes and imitates (a) thumbs up (b) Palm (c) Point up and (d) Fist gestures. In (c) robot looks towards its left to focus hand.

setup where human is interacting with robot via hand gestures.

The most important aspect in HRI is the efficiency of the whole system. In our experimentation, the system takes $70msec$ to process each image. In other words, the system processes around 15 frames per second, which is adequate for real time processing. The size of the dictionary primarily effects the processing time. However, the size of dictionary can not be reduced immensely, otherwise it would result in poor recognition rate.

C. Comparison with State-of-the-Art

The system reports 94% recognition rate for 18 different hand gestures in dynamic environments. Ren et al. [13] used finger earth movers distance metric to recognize hand gesture. They used Microsoft Kinect to capture images. For 10 gestures their recognition rate is 94% on a very limited database. Their system works good for simple gestures but gestures like thumbs up, inquire gesture etc. can not recognize efficiently as fingers are closed. Yang et al. [14] used monocular camera to recognize 18 different gestures with around 97% accuracy. The main drawback of their approach is the skin detection part which is highly unstable in illumination changes. They also put a constraint on their system that human should be present on the edge of the frame. Dardas and Georganas [9] used SIFT features and bag of words model to recognize hand using monocular camera images with 94% accuracy. Their database is extremely limited with uniform background and they also use skin detection to detect the hand. Approaches using skin detection for segmenting hand report poor accuracy when dealing with dynamic environment with illumination variation.

V. Conclusion

Human-Robot Interaction is the most important function for the emerging social robot which is able to interact with people using natural gestures. Hand gesture based interface offers a way to enable human to interact with robots more easily and efficiently. We use ASUS Xtion for hand segmentation and use bag-of-features approach using SIFT keypoints to recognize different hand gestures. The presented approach is highly efficient and invariant to cluttered backgrounds, illumination changes, slight rotation and scale. Our hand gesture interface is used by humanoid robot that recognizes gestures in $70msec$ (real time). Robot imitates gestures and also changes its face expressions according to the gestures. Our hand gesture recognition interface reports 94% recognition rate for 18 hand gestures in dynamic environments. In future, this system can be extended for both hands and can also recognize facial expressions of a human. It would help the robot to know the emotional state of human. The same approach can be used for recognizing dynamic hand gestures for recognizing different actions, e.g., clapping, jumping, punching etc.

References

[1] P. Noller, Nonverbal Communication in Close Relationships, 0th ed. SAGE Publications, Inc., 2006, pp. 403–421.

[2] Y. Huang, D. Monekosso, H. Wang, and J. Augusto, “A concept grounding approach for glove-based gesture recognition,” in Intelligent Environments (IE), 2011 7th International Conference on, July 2011, pp. 358–361.

[3] E. Marilly, A. Gonguet, O. Martinot, and F. Pain, “Gesture interactions with video: From algorithms to user evaluation,” Bell Labs Technical Journal, vol. 17, no. 4, March 2013, pp. 103–118.

[4] D.-Y. Huang, W.-C. Hu, and S.-H. Chang, “Vision-based hand gesture recognition using pca+gabor filters and svm,” in Intelligent Information Hiding and Multimedia Signal Processing, 2009. IIH-MSP ’09. Fifth International Conference on, September 2009, pp. 1–4.

[5] A. L. Barczak and F. Dadgostar, “Real-time hand tracking using a set of cooperative classifiers based on haar-like features,” in Research Letters in the Information and Mathematical Sciences, Vol. 7. Massey University, 2005, pp. 29–42.

[6] A. A. Argyros and M. I. A. Lourakis, “Vision-based interpretation of hand gestures for remote control of a computer mouse,” in In Computer Vision in Human-Computer Interaction. Springer-Verlag, 2006, pp. 40–51.

[7] I. Skrypnyk and D. Lowe, “Scene modelling, recognition and tracking with invariant image features,” in Mixed and Augmented Reality, 2004. ISMAR 2004. Third IEEE and ACM International Symposium on, Nov 2004, pp. 110–119.

[8] R. Hartanto, A. Susanto, and P. Santosa, “Preliminary design of static indonesian sign language recognition system,” in Information Technology and Electrical Engineering (ICITEE), 2013 International Conference on, Oct 2013, pp. 187–192.

[9] N. Dardas and N. D. Georganas, “Real-time hand gesture detection and recognition using bag-of-features and support vector machine techniques,” Instrumentation and Measurement, IEEE Transactions on, vol. 60, no. 11, Nov 2011, pp. 3592–3607.

[10] D. Lowe, “Object recognition from local scale-invariant features,” in Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on, vol. 2, 1999, pp. 1150–1157.

[11] J. Weston and C. Watkins, “Multi-class support vector machines,” 1998.

[12] “New Humanoid Robot tu kaiserslautern,” https://agrosy.informatik.uni-kl.de/aktuelles/details/news/neuer-humanoider-roboter/, accessed: 2016-01-25.

[13] Z. Ren, J. Yuan, J. Meng, and Z. Zhang, “Robust part-based hand gesture recognition using kinect sensor,” Multimedia, IEEE Transactions on, vol. 15, no. 5, Aug 2013, pp. 1110–1120.

[14] Z. Yang, Y. Li, W. Chen, and Y. Zheng, “Dynamic hand gesture recognition using hidden markov models,” in Computer Science Education (ICCSE), 2012 7th International Conference on, July 2012, pp. 360–365.

 ISBN: 978-1-61208-468-8