Probabilistic detection of pointing directions for human-robot interaction

Dadhichi Shukla, Özgür Erkent and Justus Piater
Intelligent and Interactive Systems, Institute of Computer Science, University of Innsbruck, Austria.
Email: {Dadhichi.Shukla, Ozgur.Erkent, Justus.Piater}@uibk.ac.at

***Abstract*—Deictic gestures – pointing at things in human-human collaborative tasks – constitute a pervasive, non-verbal way of communication, used e.g. to direct attention towards objects of interest. In a human-robot interactive scenario, in order to delegate tasks from a human to a robot, one of the key requirements is to recognize and estimate the pose of the pointing gesture. Standard approaches rely on full-body or partial-body postures to detect the pointing direction. We present a probabilistic, appearance-based object detection framework to detect pointing gestures and robustly estimate the pointing direction. Our method estimates the pointing direction without assuming any human kinematic model. We propose a functional model for pointing which incorporates two types of pointing, finger pointing and tool pointing using an object in hand. We evaluate our method on a new dataset with 9 participants pointing at 10 objects.**

I. Introduction

In the field of autonomous robotic systems, social robots are increasingly becoming more relevant in our daily lives. The transition of robots from industry to domestic environments raises a fundamental challenge in developing intuitive communication mechanisms for human-robot interaction. Many human interactions involve verbal and/or non-verbal exchange of information. Amongst the various communicative actions, pointing gestures are very natural and intuitive. They are effective even in noisy environments and useful for simple robot commands, as demonstrated by Richarz et al. [1].

In this work we target assistive robots with the capability to recognize pointing gestures. In human-robot interaction, camera-based vision – being non-invasive and prop-free – is considered to be the only natural method of capturing gestures. Directing attention towards an object can be achieved by two types of pointing: (1) *finger* pointing, and (2) *tool* pointing using an elongated object in hand, as shown in Fig. 1. One of the usages of the pointing gesture is target selection, i.e. to indicate and select an object in the real world.

A. Motivation

Previous work has shown [2], [3], [4] that the line-of-sight between face and hands provide a very reliable estimate for the pointing direction. However, to compute the best estimates of the pointing direction these approaches rely primarily on the output of the preceding face, arm and hand pose estimation steps. In fact, an intrinsic part of the system for those methods requires either the user's full-body or partial-body to be completely visible in the field of view of the camera. Moreover, the subject is confined to align the pose of its face, arm and hand in accordance to the desired pointing direction. This is not the case with human-human interaction, where the pointing direction can be implicitly determined from the pose of the hand. In addition to this, providing the liberty to users from the aforementioned constraint would significantly increase the functionality of pointing gesture recognition in human-robot interactions [5], [6]. We propose a probabilistic appearance-based approach to estimate the pose of a pointing gesture and the location of the targeted object in the real world. The proposed framework is independent of the user's body posture i.e. pose of face, arm or torso, and therefore enables human-robot interaction in close proximity.

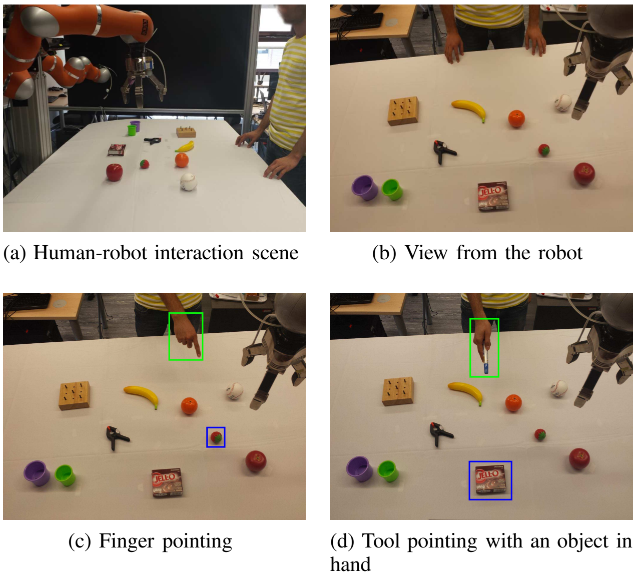

(a) Human-robot interaction scene
(b) View from the robot
(c) Finger pointing
(d) Tool pointing with an object in hand

Fig. 1: Pointing gesture recognition and target estimation.

B. Related Work

One of the natural, non-contact solutions to detect pointing gesture in the early years of human-robot interaction was proposed by Cipolla et al. [7] for pick-and-place operations. Cipolla et al. used stereo vision with active contours to track the position and pointing direction on the robot's two dimensional workspace. Since then and thanks to the latest developments in vision technology, the interest in non-contact human-computer interfaces has increased.

In previous works, the pose of the user's body was an essential requirement for the system. The method proposed by Hosoya et al. [8] computes the pointing direction by estimating the 3D position of the shoulders and the arms from depth and stereo color images. Richarz et al. [1] developed a robot guidance method based on a Multi-Layer Perceptron (MLP) which was tested with low cost monocular webcams. Watanabe et al. [9] proposed a multi-camera system (8 cameras) to estimate the direction of pointing preceded by face and hands detection. Jojic et al. [10] proposed a pointing gesture detection based on depth information to overcome the limitations of color-based approaches which are sensitive to lighting changes. The work proposed by Stiefelhagen et al. [2] is an HMM-based approach to estimate 3D position of the face and hands using color and disparity information. However, the pointing gesture detection is preceded by three subsequent feature sequences. Park et al. [3] also proposed an HMM-based approach in two stages to recognize the pointing gesture. Park et al.'s method was the first one to include the scale of gestures, however, due to the number of HMM states this method depends on a large training data set consisting of more than 1500 samples and, thus, it is computationally expensive. The research conducted by Sato et al. [11] uses a pair of orthogonally placed cameras to recognize pointing gestures based on Fuzzy neural networks. The method proposed by Hu et al. [12] used an adapted Adaboost cascade detector [13] for hand gesture detection and an active appearance model to estimate direction from two orthogonal views. The hand detection is preceded by wrist estimation. The method proposes two detectors trained for two views separately with more than 100 positive samples and more than 140 negative samples for each view. More recently, Pateraki et al. [4] introduced visual estimation of pointed targets for robot guidance via fusion of the pose of the face and hand orientation.

Regarding methods independent of human posture, McGuire et al. [5] proposed close range pointing to objects on a table using stereo images. They used a multi-layer perceptron classifier to localize hand and finger tips and estimate the location of the targeted object from the finger direction. Another approach, independent of human-body pose was proposed recently by Fujita et al. [6] to recognize mid-air pointing. They calculate classification scores in a sliding window for hand postures with different pointing directions. The classification scores are interpolated to detect pointing gestures and estimate its direction. However, the method heavily depends on a large number of positive training samples about 65000 and about 1500 negative samples. Our work is also independent of human posture as the aforementioned studies. One important difference with the former is that in [5] authors use scaffolding where the human operator guides the robot to teach what to focus through speech command and hand-gesture. Unfortunately, having such a dedicated human operator is a serious constraint. And the later method [6] mainly focuses on pointing by moving index finger and does not address gesture recognition in different poses of the hand.

We briefly summarize the previous approaches and the proposed work in Table I reviewing errors in the estimated pointing direction.

C. Contribution

The main challenge in pointing gestures is that it can provide only a coarse spatial information of the targeted object. Thus, there is always an intrinsic ambiguity in the precise 3D location when pointing. Stiefelhagen et al. [2] asserted that it is possible to disambiguate possible pointing targets with an average error of less than 20° at a distance of about 2.5 meters.

Methods	N	e_a	e_d	B	S
Cipolla et al. [7]	2	-	2cm*	×	×
Hosoya et al. [8]	2	-	0.5-1m	✓	×
Jojic et al. [10]	2	-	15cm	✓	×
Stiefelhagen et al. [2]	2	20°	-	✓	✓
Park et al. [3]	4	7.2-18.7°	-	✓	×
Richarz et al. [1]	1	10°	59cm	✓	×
Watanabe et al. [9]	8	2.14°	-	✓	×
Sato et al. [11]	2	-	22.7cm	✓	×
Hu et al. [12]	2		91%**	✓	×
Pateraki et al. [4]	1		85.235%**	✓	×
Fujita et al. [6]	2		69.585%**	×	×
Our method	2	10°	93.45%**	×	×

TABLE I: Summary of pointing gesture recognition systems. Previous works are summarized based on following parameters: Number of cameras (N), Angular error (e_a), Distance error (e_d) i.e. error between the ground truth of the target location and the estimated 3D location, Full-body or partial-body pose essential (B), Multi-modal i.e use of speech (S). *On 40 cm workspace. ** Average pointing gesture recognition rate.

In order to take steps forward for overcoming this challenge, we adopt the probabilistic appearance-based object detection and pose estimation framework proposed by Teney et al. [14], [15]. The method works on 2D images and is applicable to various types of image features. A brief description of the approach and its extension is presented in section II-A. In this work, we aim to make human pointing gestures accurate enough for a robot to estimate the pointing direction in the real world.

Our test scenario is to point at household objects placed on a table, albeit this can be applied to any unstructured task. The main contributions of this work are:

- To use a probabilistic appearance-based model [15] to detect pointing gestures and estimate its direction. The proposed framework incorporates the variability of hands concerning color, size and scale.
- Human-robot interaction in close vicinity without any prior initialization of the scene or knowledge about the human body pose.
- Two types of pointing gestures are considered: (1) finger pointing, and (2) tool pointing with elongated object in hand.
- A corpus of 180 pointing gestures with ground truth acquired by a RGB-D sensor with 9 participants pointing at 10 objects on a table.

II. Pointing Gesture Detection and Pose Estimation

The human hand is deformable by nature and very diverse in shape, size and color among different people, which makes hand pose estimation an interesting and challenging research topic. We address it by adopting a probabilistic appearance-based object recognition and pose estimation framework [14], [15]. The method accommodates variability in scale, shape and appearance of objects. We use this capability in order to

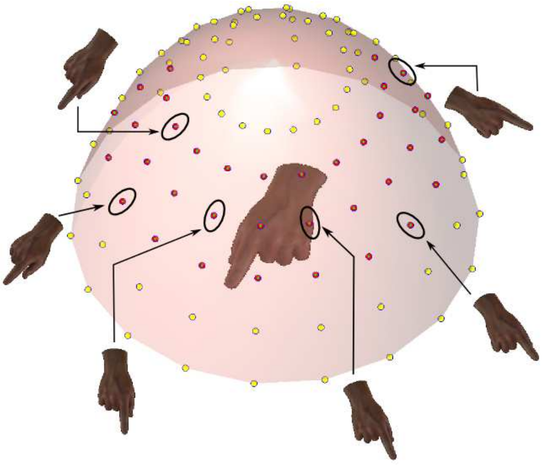


Fig. 2: Training images captured from different viewpoints. The viewpoints marked *red* are a subset of the training data used to learn the model of the pointing gesture.

train the framework with a synthetic pointing gesture model. We extend *Teney et al.'s work* based on image gradients with additional features like depth range, hue color and surface normals in order to improve the accuracy of the detection. The synthetic hand model used to generate the training data was proposed by Romero et al. [16], [17]. An illustration of a set of training viewpoints on the viewing sphere and few of the sample training images are shown in Fig. 2.

A. Probabilistic Appearance-based Model

The object detection and pose estimation framework we use is based on probabilistic representation of both the training and the test data. The method can be seen as a smoothing over the available data, providing continuous distributions of features and interpolating, to some extent, between the available data points.

Learning object models – Pose-Appearance space: We start by extracting different types of features from a given image I, each type denoted by an index: $f = 1, \ldots, F$, i.e. edge points and orientation, depth range, hue color and surface normals. Each feature x is defined by its position $p_x \in \mathbb{R}^2$ in the image and its appearance attributes a_x. The appearance attributes include local orientation ($\in S_1^+ = [0, \pi]$) of an edge point, depth values ($\in \mathbb{R}^+$ in meters) of the object, hue values ($\in [0, 1]$), and the surface normals ($\in S_1^+ = [0, \pi]$). An illustration of the features used is shown in Fig. 3.

Therefore, for each type f of features, a set $I^f = \bigcup_i^{\|f\|} x_i^f$ is generated. Each feature is characterized by its position and appearance $x_i^f =< p_{x_i}^f, a_{x_i}^f >$, with $x_i^f \in A^f$, where A^f is the appearance space of image features. For each type of image features f, we use the set of features I^f to define a distribution given by

$$\phi^f(x^f) = \sum_{x_i^f \in I^f} \mathrm{w}(x_i^f) \mathcal{N}(p_{x_i^f}; \mathrm{p}_x^f, \sigma_p) \mathrm{K}^f(a_{x_i}; \mathrm{a}_x), \quad (1)$$

where, $\mathcal{N}$ is a Gaussian kernel for the positions of the features, K^f is a kernel for their appearance, and $\mathrm{w}(x_i^f)$ is the weight of the feature x_i i.e., $\mathrm{w}(x_i) = 1/\|I^f\| \; \forall x_i \in I^f$.

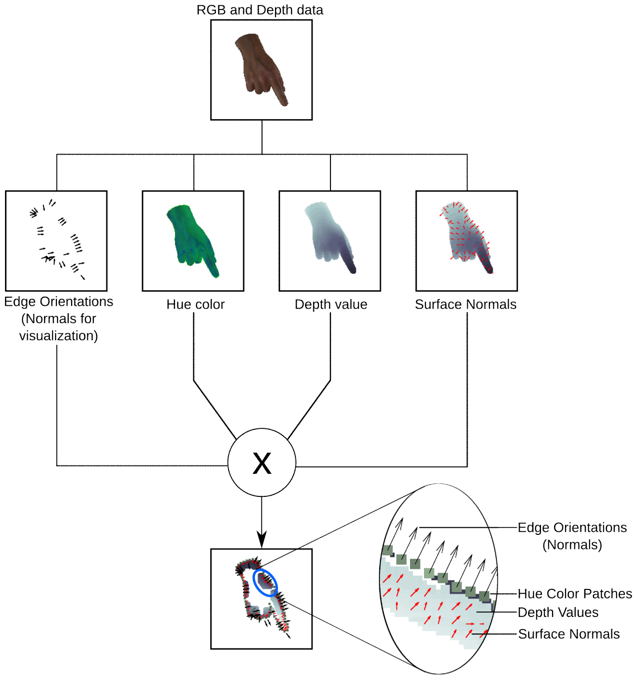


Fig. 3: Different types of features used to create the *pose/appearance* space.

We obtain two distributions, one $\phi_{I_{train}}^f$ from the set of training images (I_{train}) where each image correspond to one viewpoint $v \in S^2$. Another distribution $\phi_{I_{test}}^f$ corresponds to the images used for testing the learned model, where viewpoint is of course, unknown [14].

Pose Inference: A training view can be observed in the test image with a similarity transformation t (in-plane translation, rotation and scaling), applied by a function $\mathcal{T}_t(x)$. The similarity between the test and the training images can be obtained by cross-correlation of the training and test distributions

$$\left(\phi_{test}^f \star \phi_{train_v}^f\right)(t) = \int_{A^f} \phi_{test}^f \phi_{train_v}^f(\mathcal{T}_t(x)) \mathrm{d}x, \quad (2)$$

where, v is one of the viewpoint. To increase the efficiency of the system, samples are drawn from the test and training images by Monte Carlo integration [18]. An efficient approximate evaluation of Eq. 2 is achieved by drawing L particles from test distribution and L' particles from the training distribution.Then the cross-correlation for one feature type f becomes

$$\left(\phi_{test}^f \star \phi_{train_v}^f\right)(t) \approx \frac{1}{LL'} \sum_i^L \sum_j^{L'} \mathrm{w}(x_j) \mathcal{N} \quad (3)$$
$$(p_{x_i}; \mathcal{T}_t(p_{x_j}), \sigma_{p_{x_j}}) \mathrm{K}^f(a_{x_i}; a_{x_j}).$$

The full similarity measure between test and training images is the product over f in Eq. 3, taking into account all image features is given by

$$s_{\text{test,train}_v}(t) = \prod_f (\phi_{\text{test}}^f \star \phi_{\text{train}_v}^f)(t). \quad (4)$$

The result $s_{\text{test,train}_v}(t)$, from now on, s for convenience, represents the scores associated with the possible set of poses

$\mathcal{S}$ of the training images recognized in the test scene. The estimated poses in $\mathcal{S}$ represent 6 degrees of freedom of the object, which can be transformed from the image space to the robot space. Therefore, the estimated pose of the hand also indicates the direction of the pointing gesture.

B. Image Features and Kernel Definition

In this section we describe the image features used in this work and provide their kernel definitions. Each image feature is associated with a kernel K^f as shown in Eq. 1. The appearance of the image features is defined such that $a^f_{x_i}, a^f_{x_j} \in \mathcal{A}^f$. The normalization coefficients C_j's and the concentration parameters analogous to the inverse variance of Gaussian distribution κ_j's in following definitions varies according to the type of the image feature, where $j = \{1, 2, 3, 4\}$. If any of the following features is absent at a certain point, it is not considered for the appearance model. The description of the kernel for different feature types used in this work is as follows

a) Edge Orientation: It is the local orientation (an angle in $S^+_1 = [0, \pi]$) of the edge point at position p_x in the image. It is defined as $\mathcal{A}^{ori} = \mathbb{R}^2 \times S^+_1$. The kernel associated with this attribute uses the von Mises distribution [19] on the half circle as

$$K^{ori}(a^{ori}_{x_1}, a^{ori}_{x_2}) = C_1 \mathrm{e}^{\kappa_1 \cos(a^{ori}_{x_1} - a^{ori}_{x_2})}.$$

b) Depth Value: It is the depth value of points on the surface of the object i.e. distance (in meters) between the camera and the object in the real world. Depth feature is defined as $\mathcal{A}^{dep} = \mathbb{R}^+$. The kernel associated with the depth feature type is given by

$$K^{dep}(a^{dep}_{x_1}, a^{dep}_{x_2}) = C_2 \mathrm{e}^{\kappa_2 (a^{dep}_{x_1} - a^{dep}_{x_2})^2}, \forall a^{dep}_{x_1}, a^{dep}_{x_2} \in \mathbb{R}^+.$$

In the absence of the depth feature due to transparent or reflective nature of the object material the method gives priority to the use of other features (e.g. orientation, color).

c) Hue Color Values: It is the hue color value of the object defined as $\mathcal{A}^{hue} = \mathbb{R}^+ \in [0, 1]$. The kernel for hue feature type uses the von Mises distribution given by

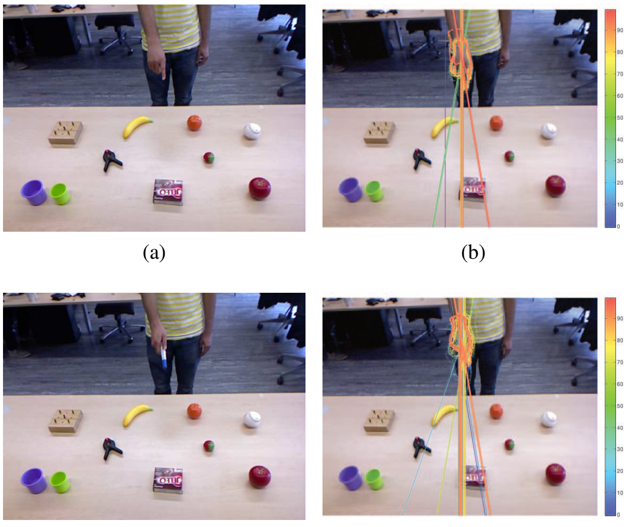
$$K^{hue}(a^{hue}_{x_1}, a^{hue}_{x_2}) = C_3 \mathrm{e}^{\kappa_3 \cos(a^{hue}_{x_1} - a^{hue}_{x_2})}, \forall a^{hue}_{x_1}, a^{hue}_{x_2} \in [0, 1].$$

d) Surface normals: It is the normal to the point on the surface of the object. The normal vector at a point p is given by n, where its attribute is the partial derivative of this surface normal. It is defined as $\mathcal{A}^{nor} = \mathbb{S}_2$. The kernel for surface normals uses the von Mises distribution given by

$$K^{nor}(a^{nor}_{x_1}, a^{nor}_{x_2}) = C_4 \mathrm{e}^{\kappa_4 cos(a^{nor}_{x_1} - a^{nor}_{x_2})}, \forall a^{nor}_{x_1}, a^{nor}_{x_2} \in \mathbb{S}^2.$$

C. Estimate of the targeted Object in 3D

The probabilistic appearance-based method results in a set $\mathcal{S}$ of the estimated 6D poses of the trained object recognized in a test scene as given by Eq. 4. Therefore, the pose of the pointing gesture is the 3D direction d associated with a score s. The direction d is simply the line along the estimated pose in 3D space. If the pose of the objects is known the targeted object can be localized with a distance measure. Each object is associated with a confidence measure based on its distance D to the estimated directions and the score of the pointing direction. The confidence of the object is given by


(a)

(b)

(c)

(d)

Fig. 4: Target object estimates in the real world.

$$p^*_j = \frac{1}{\text{argmin} \prod_{i=1}^{m} f(D_{ij}, s_i)}, \tag{5}$$

where, m is the number of estimated directions of the j'th object. The goal function is given by

$$f(D_{ij}, s_i) = \frac{1}{e^{s_i}} (||o_j - d_i||), i = 1, \ldots, m, \tag{6}$$

where, o_j is the location of the j'th object in real world. It can be seen in Fig. 4 that the estimated directions do not point at a single target. Here, the directions are color coded with *red* being the *best estimate* (the one with the highest score). The object with the highest confidence measure is considered as the targeted object.

III. Experimental Setup

Our pointing gesture detection and pose estimation framework can be trained with a few number of training images captured from different viewpoints. The system learns the model of pointing gestures from synthetic training images generated using *libhand* [17]. The learned model is used to detect pointing gesture of different participants with variability in hand sizes and shapes.

It is difficult to perform a quantitative evaluation of pointing direction because of the lack of ground truth. To overcome this problem, we recorded pointing gestures from nine people pointing at ten objects placed on a table using a Kinect RGB-D camera with known ground truth. Two types of pointing were recorded in the process: (1) finger pointing, and (2) tool pointing with an object in hand. Participants used a whiteboard marker for tool pointing. Please note that there is no information of the object in hand in the learned model. The participants were standing approximately at a distance of 1.35m from the camera. No specific instructions were given to participants except the sequence of objects to be pointed at, allowing them to point in a natural fashion. This dataset is first of a kind to evaluate short range pointing gesture detection and direction estimation. Our method is independent of full-body or partial-body pose, therefore, it is possible to perform human-robot interaction in close proximity.

The participants pointed at objects placed on a planar surface (table), pointing directions were within the range of -40° to 30° horizontally (azimuth angles) and within the range of 75° to 105° vertically (elevation angles). We created a novel dataset consisting of 39 training images and 180 pointing gesture scenarios. Both, training and test sets are provided with manually labelled ground truth. The 39 training images are used to create the *pose/appearance* space (section II-A). 12 images (6 finger pointing and 6 tool pointing) out of the 180 test images are used to tune the parameters for the kernel definitions (section II-B). The remaining test scenarios (168) are used for evaluation.

The objects used in this work are from Yale-CMU-Berkeley (YCB) Object and Model set [20]. The objects in the set cover a wide range of aspects of robot manipulation. It includes objects of daily life that can be present in any human environment. The objects vary in shape, size, texture, weight and rigidity.

IV. Results and Evaluation

To evaluate our method, we compared the estimated pointing direction with the manually labelled ground truth. The ground truth of the direction is computed using the depth data which provides the pose (location, azimuth angle and elevation angle) of the hand, and location of the object (i.e. its centroid) in the real world. The pointing gesture detection system results in a set $\mathcal{S}$ of pose hypotheses each of them associated with a score s as described in section II-A. To speed up the estimation of the pointing direction we mask out the edges of the background (floor, table, chair, etc.) using depth information. It is to be noted that no other masking strategies (skin segmentation or hand detection) were used to eliminate any other hand present in the image. Moreover, our framework is trained with the model of a pointing gesture; therefore non-pointing hands are unlikely to be detected.

We performed two sets of experiments. The first is aimed at evaluating the accuracy and standard deviation of the highest-scoring pointing direction as well as the detection with minimum error in pose angles. Then, we performed a quantitative analysis on the pose error of the 168 test images. A hypothesis is considered correct if it meets a given error tolerance and a minimal required hypothesis score.

A. Experiment 1: Accuracy and Standard deviation analysis

The training images used to learn the pointing gesture model are spaced at 10° intervals in pose angles (azimuth and elevation). For evaluation we calculate the mean error μ and the standard deviation σ of the following two pose estimates: (1) the best estimated direction i.e. highest-scoring pointing direction, and (2) the estimated direction with minimum error in pose angles i.e. nearest estimate. Table II shows that our method exhibits low systematic error and standard deviations of around 10° in both pose angles for the *best estimates* (i.e. the highest-scoring pointing direction). If we consider the estimated directions with minimum error (*nearest estimates*), our method has a standard deviation of only around 4° and 3° in azimuth and elevation, respectively. In addition to this, we see the effect of the score s in the pointing direction provides a more accurate estimation. The score is included as the weight given by $w = 1/e^s$ to calculate the weighted mean μ_s and the standard deviation σ_s. Errors in the estimated pose angles reduced significantly when the score is included.

	Highest-scoring estimate		Nearest estimate	
	ϕ	θ	ϕ	θ
μ	-0.0628	-0.6689	0.1739	-0.0180
σ	10.5475	10.0189	4.0765	3.3029
μ_s	-0.0231	-0.2461	0.1226	-0.0489
σ_s	3.8802	3.6857	3.6773	3.0824

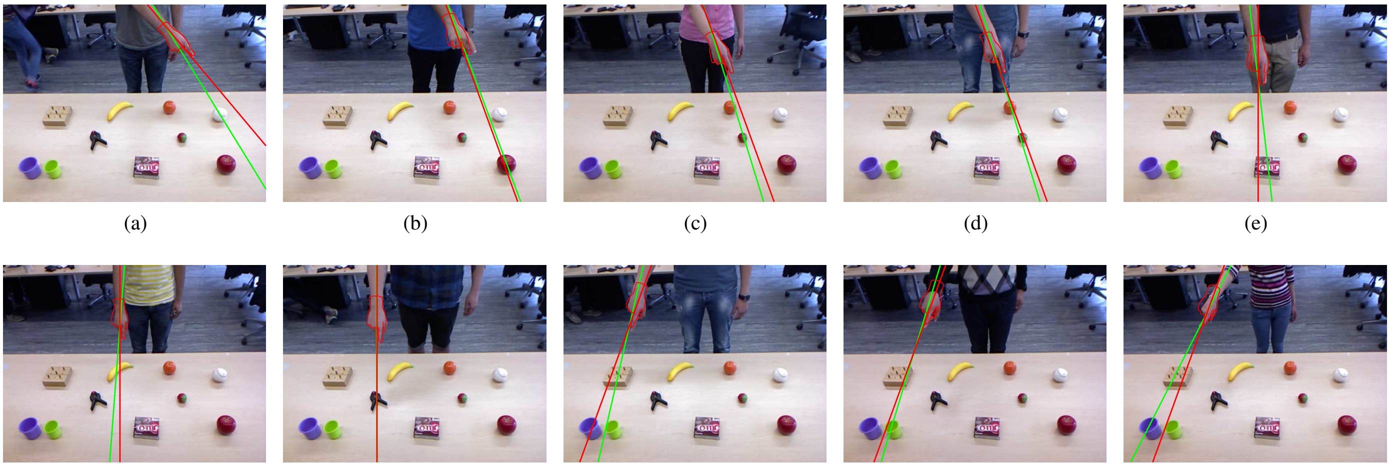
TABLE II: Mean error and standard deviation in degrees. ϕ - Azimuth angle, θ - Elevation angle.

Figures 5 and 6 are examples of the best estimate compared to the ground truth. The estimated direction is marked with a *red* line and the ground truth is marked in *green*. The results shown are 2D illustrations of the 3D pose. Our method can accurately estimate the direction of the pointing gestures despite the variation of sizes and scales of the hand. The learned model closely fits various pointing gestures. Moreover, due to the similarity in the appearance of the finger pointing and tool pointing, our method can estimate the tool pointing using the same learned model. Please note that our method is independent of the pose of other body parts, the user is not constrained to learn specific ways of pointing like aligning forearm pose with pointing or looking in the direction of the targeted object as in other works [2], [3], [4], [11]. The errors in estimated pointing directions are low enough for human-robot interactive tasks in close proximity.

The ambiguity (defined as the distance from the ground truth location to the estimated target location in pointing gesture) inherent in pointing gestures can be seen in Figs. 5i and 5j, where participants are pointing at the *green* cup and the *purple* cup, respectively. Since the objects are less than 10° apart, our method shows certain ambiguity in detection. This ambiguity can be reduced by taking the remaining hypotheses into consideration. We show in Fig. 7 some cases where our method fails due to high but spurious similarities between the appearance of a training model and the hand of the participant. For example, in Figs. 7a, 7b and 7c the model inaccurately recognises the *thumb* as the index finger.

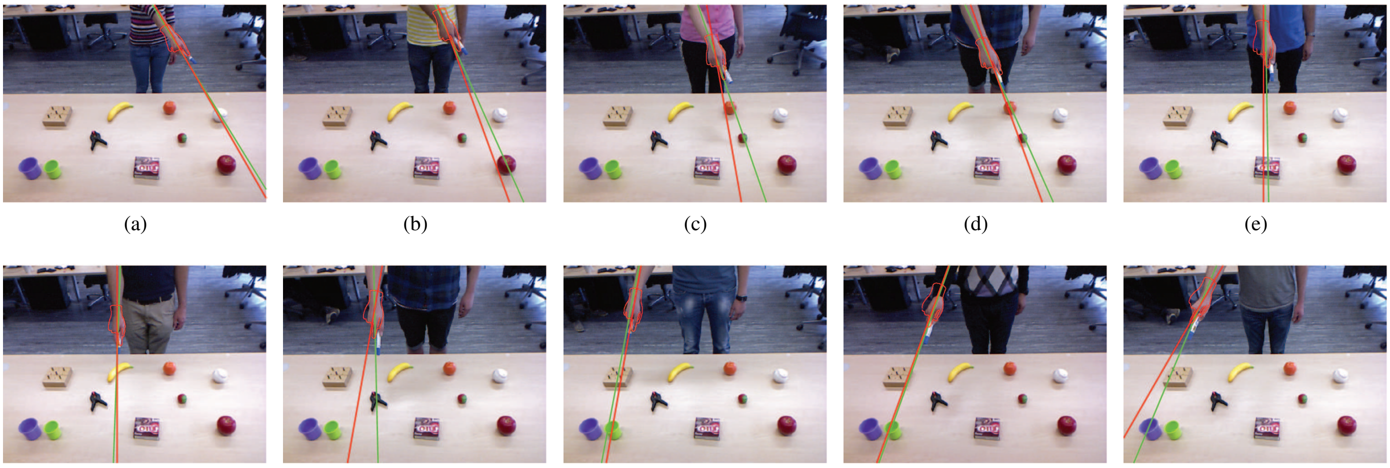
B. Experiment 2: Quantitative Analysis

We perform quantitative analysis on 168 pointing test images with 9 participants. The effectiveness of the proposed work was verified by counting a detected pointing direction as correct if the error between the estimated pose is and the ground truth is within a defined permissible range. Additionally, to evaluate the robustness of our method we analyse the number of true directions in the set $\mathcal{S}$ with a minimal required hypothesis score over total number of detections. We varied two parameters: (1) The average error e between the pose angle estimation and the ground truth, and (2) a minimal hypothesis score threshold which is given by factor t of the score of


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

(f) (g) (h) (i) (j)

Fig. 5: Finger pointing direction estimation for human-robot interaction. *Color code:* Red – Best estimated pointing gesture and 3D pointing direction, Green – Ground truth direction in 3D.


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

(f) (g) (h) (i) (j)

Fig. 6: Tool pointing direction estimation using a whiteboard marker in hand for human-robot interaction. *Color code:* Red – Best estimated pointing gesture and 3D pointing direction, Green – Ground truth direction in 3D.

the best estimate (i.e. highest score). The pose estimate is considered as a true detection if the score of the estimated direction s_{est} is above score threshold i.e. $s_{est} > t \cdot s_{best}$. The accuracy plot with varying e and t is shown in Fig. 8. The factor t controls the accuracy rate at a defined e. If the factor t is reduced hypotheses with lower scores are accepted as true estimates.

When the best pointing directions i.e. $t = 1.0$ are considered, we achieve an accuracy of 93.45% considering an error in the range of $[-10°, 10°]$. At $t = 0.2$ the accuracy rate increases to 97.62% which shows that hypotheses with lower scores also accurately estimates the pointing direction. With the direction error within the range of $[-15°, 15°]$ and $t = 1.0$ we achieve a accuracy rate of 99.40%.

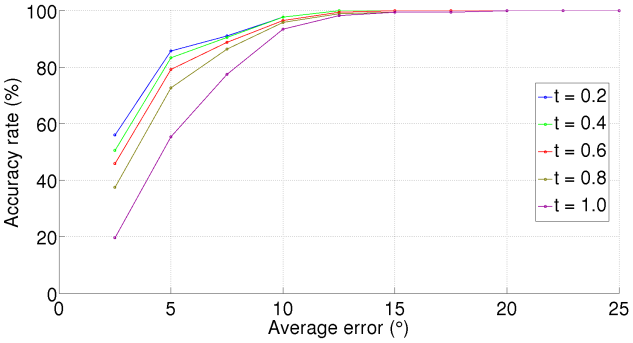


Fig. 8: Accuracy rate of the pointing direction at different error range and score thresholds.

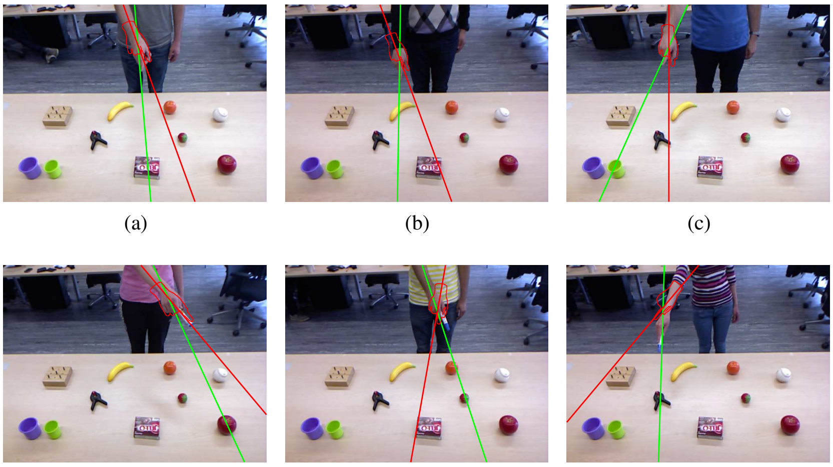


Fig. 7: Failed estimations in pointing direction. The errors in estimated 3D pose and the ground truth here are greater than 20°. *Color code:* Red – Best estimated pointing gesture and 3D pointing direction, Green – Ground truth direction in 3D.

V. Conclusion and Future work

For a robot to work in a close proximity of humans it is necessary to detect human gestures independent of human-body pose. Pointing at the objects to direct the attention of others, is ubiquitous in non-verbal human communication. The meaning of the pointing gesture must be unambiguous to a robot. We have presented a novel probabilistic appearance-based approach to detect pointing directions for human-robot interaction. The proposed framework can accommodate variability in size and scale of the pointing gesture. This endows the proposed method to detect pointing gestures by various participants with a single hand model.

The following conclusions can be drawn from this study:

- A pointing gesture model can be learned with just a few training images. The learning phase does not require negative samples.
- One model fits multiple pointing gestures at various scales and sizes. The similarity in appearance of pointing makes it convenient to detect pointing with elongated objects in hand.
- Our method is independent of any initialization phase or human-body pose. Therefore, the user can point freely without any constraints or instructions.
- A new pointing direction dataset with ground truth, which will be made publicly available in the near future.

Our test results show that our method achieves an exceptionally good accuracy rate in the estimation of the pointing direction without any constraint imposed on the user. Although, in some test scenarios the method failed to accurately estimate the pointing direction. We are currently working on not just using the highest score hypothesis but a combination of high score hypotheses into a probabilistic framework that can provide with a more accurate hand pose and thus, a better estimate of its pointing direction.

Acknowledgement

The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 (Specific Programme Cooperation, Theme 3, Information and Communication Technologies) under grant agreement no. 610878, 3rd HAND. We would like to thank Dr. Antonio Rodriguez Sanchez (University of Innsbruck) for discussions and suggestions.

References

[1] J. Richarz, A. Scheidig, C. Martin, S. Müller, and H.-M. Gross, "A monocular pointing pose estimator for gestural instruction of a mobile robot," *International Journal of Advanced Robotic Systems*, vol. 4, no. 1, pp. 139–150, 2007.

[2] R. Stiefelhagen, C. Fügen, P. Gieselmann, H. Holzapfel, K. Nickel, and A. Waibel, "Natural human-robot interaction using speech, head pose and gestures," in *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, vol. 3. IEEE, 2004, pp. 2422–2427.

[3] C.-B. Park and S.-W. Lee, "Real-time 3d pointing gesture recognition for mobile robots with cascade hmm and particle filter," *Image and Vision Computing*, vol. 29, no. 1, pp. 51–63, 2011.

[4] M. Pateraki, H. Baltzakis, and P. Trahanias, "Visual estimation of pointed targets for robot guidance via fusion of face pose and hand orientation," *Computer Vision and Image Understanding*, vol. 120, pp. 1–13, 2014.

[5] P. McGuire, J. Fritsch, J. J. Steil, F. Röthling, G. A. Fink, S. Wachsmuth, G. Sagerer, and H. Ritter, "Multi-modal human-machine communication for instructing robot grasping tasks," in *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, vol. 2. IEEE, 2002, pp. 1082–1088.

[6] D. Fujita and T. Komuro, "Three-dimensional hand pointing recognition using two cameras by interpolation and integration of classification scores," in *Computer Vision-ECCV 2014 Workshops*. Springer, 2014, pp. 713–726.

[7] R. Cipolla and N. J. Hollinghurst, "Human-robot interface by pointing with uncalibrated stereo vision," *Image and Vision Computing*, vol. 14, no. 3, pp. 171–178, 1996.

[8] E. Hosoya, H. Sato, M. Kitabata, I. Harada, H. Nojima, and A. Onozawa, "Arm-pointer: 3d pointing interface for real-world interaction," in *Computer Vision in Human-Computer Interaction*. Springer, 2004, pp. 72–82.

[9] H. Watanabe, H. Hongo, M. Yasumoto, and K. Yamamoto, “Detection and estimation of omni-directional pointing gestures using multiple cameras.” in *MVA*, 2000, pp. 345–348.

[10] N. Jojic, B. Brumitt, B. Meyers, S. Harris, and T. Huang, “Detection and estimation of pointing gestures in dense disparity maps,” in *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*. IEEE, 2000, pp. 468–475.

[11] E. Sato, T. Yamaguchi, and F. Harashima, “Natural interface using pointing behavior for human–robot gestural interaction,” *Industrial Electronics, IEEE Transactions on*, vol. 54, no. 2, pp. 1105–1112, 2007.

[12] K. Hu, S. Canavan, and L. Yin, “Hand pointing estimation for human computer interaction based on two orthogonal-views,” in *Pattern Recognition (ICPR), 2010 20th International Conference on*. IEEE, 2010, pp. 3760–3763.

[13] P. Viola and M. J. Jones, “Robust real-time face detection,” *International journal of computer vision*, vol. 57, no. 2, pp. 137–154, 2004.

[14] D. Teney and J. Piater, “Multiview feature distributions for object detection and continuous pose estimation,” *Computer Vision and Image Understanding*, vol. 125, pp. 265–282, 8 2014. [Online]. Available: https://iis.uibk.ac.at/public/papers/Teney-2014-CVIU.pdf

[15] ——, “Generalized Exemplar-Based Full Pose Estimation from 2D Images without Correspondences,” in *Digital Image Computing: Techniques and Applications*. IEEE, 2012. [Online]. Available: https://iis.uibk.ac.at/public/papers/Teney-2012-DICTA.pdf

[16] J. Romero, H. Kjellström, and D. Kragic, “Monocular real-time 3d articulated hand pose estimation,” in *Humanoid Robots, 2009. Humanoids 2009. 9th IEEE-RAS International Conference on*. IEEE, 2009, pp. 87–92.

[17] M. Šarić, “Libhand: A library for hand articulation,” 2011, version 0.9. [Online]. Available: http://www.libhand.org/

[18] R. E. Caflisch, “Monte carlo and quasi-monte carlo methods,” *Acta numerica*, vol. 7, pp. 1–49, 1998.

[19] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.

[20] B. Çalli, A. Walsman, A. Singh, S. Srinivasa, P. Abbeel, and A. M. Dollar, “Benchmarking in manipulation research: The YCB object and model set and benchmarking protocols,” *CoRR*, vol. abs/1502.03143, 2015. [Online]. Available: http://arxiv.org/abs/1502.03143