Prediction of user's grasping intentions based on eye-hand coordination

Miguel Carrasco and Xavier Clady

***Abstract*— Eye-hand coordination is a primordial reach-to-grasp action performed by a human hand when reaching for an object. This paper proposes the use of a visual sensor which allows the simultaneous analysis of hand and eye motions in order to recognize the reach-to-grasp movement, i.e. to predict the grasping gesture. This solution fuses two viewpoints taken from the user's perspective. First, by using an eye-tracker device attached to the user's head; and second, by utilizing a wearable camera attached to the user's hand. The information from these two viewpoints is used to characterize multiple hand movements in conjunction with eye-gaze movements through a Hidden-Markov Model framework. In various experiments, we show that combining these two sources of information allows the prediction of a reach-to-grasp movement as well as the desired object.**

keywords: **Visual system, gesture recognition, eye-hand coordination, reach-to-grasp movement, object recognition.**

I. Introduction

Over the past few years, there has been a growing interest in the design of robot-based rehabilitation therapy to aid patients with arm disabilities, specifically in those with neurological injuries or disorders (e.g. [1], [2], [3]). The majority of these studies have been dedicated to the development of complex active-orthosis systems to improve and reinforce upper limb motion (e.g. [4], [3]). Other investigations have advocated the analysis of human motion in order to infer human actions (eg. [5], [6]). Along that same line, the system proposed by Jarrasé, et al. [7] shows how human motion prediction can be used to improve robot transparency[1]. Nonetheless, in order for these systems to be applied in realistic situations, they need to be able to identify early on which gesture will be used; in other words, what are the user's intentions? Thus, robot controllers could produce a better response in comparison to purely reactive strategies [7].

This paper investigates a novel approach to recognize human intentions by fusing information from multiple wearable visual devices. With the goal of designing an easily embedded device for an active orthosis, we developed a method that utilizes only the user's gaze and reach-to-grasp movements. The system is composed of 1) an eye-tracker device that captures a field-of-view (FOV) similar to the user's as well as the user's estimated gaze position with regards to the camera; and 2) a camera attached to the user's wrist that captures a similar scene to that of the eye-tracker camera (Fig.1). The visual information captured from these two points-of-view allows us to exploit eye-hand coordination through an HMM process to predict user grasp intentions and the desired object. The methodology is composed of two main steps. Firstly, we propose a method that uses only the visual information from the wrist camera to recognize reach-to-grasp movements. Secondly, this information is combined with an object recognition methodology and eye-movement analysis in order to differentiate fixations from reach-to-grasp movements. This method is capable of detecting when a user wants to grasp an object as well as which specific object is desired from the scene. The experiments have been carried out in two stages according to the each phase of our methodology. The objects have been placed on a fixed table in similar conditions to a classic therapy protocol [8].

The rest of the paper is organized as follows: Section II discusses prior work on human gesture recognition; Section III explains the proposed method; Section IV shows experimental results; and finally, Section V presents our contributions and succinctly describes some ongoing and future works.

II. Related Work

Hand-Eye coordination in grasping gestures: Human beings posses a highly developed ability to grasp objects under many different conditions taking into account variations in position, location, structure and orientation. This natural ability controlled by the human brain is called *eye-hand coordination*. This action is regulated by the interaction of several sensorimotor systems such as the visual system, vestibular system and proprioceptive system [9]. In this investigation we define the grasping intention as the active

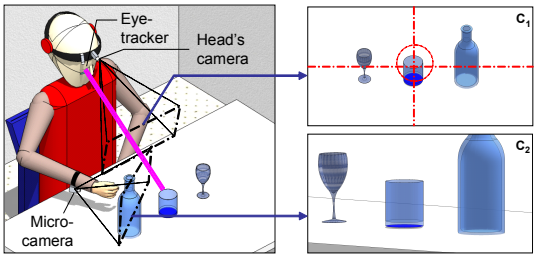


Fig. 1. User's posture when performing a reach-to-grasp action using an eye-tracker and a camera beneath the wrist

This work was supported by the Comisión Nacional de Investigación Cientifica y Technólogica (CONICYT), Chile. (Fondecyt grant. no. 11100098)

Miguel Carrasco is with the Escuela de Ingeniería Informática, Facultad de Ingeniería, Universidad Diego Portales, Av. Ejército 441, Santiago, Chile `miguel.carrasco@mail.udp.cl`

Xavier Clady is with the Institut des Systèmes Intelligents et de Robotique (ISIR), Université Pierre et Marie Curie-UPMC, CNRS UMR 2777. France `xavier.clady@upmc.fr`

[1]the robot's capacity to follow human movements without any human resistive force

conscious action of reaching for an object. The gaze remains fixed for a short time when the user initiates a reach-to-grasp action towards the object, simultaneously, hand trajectory remains stable while moving toward the object [10]. These two features are essential when predicting user intention.

Human motion analysis by computer vision methods: These methods can be divided into three main approaches: Passive, Wearable and Pointer paradigm which are relative to user and camera positioning. The Passive approach (see recent surveys [11], [12]) has two main scenarios depending on whether the subject is captured with just one stationary camera or with multiple cameras from multiple perspectives in correspondence. The Wearable approach uses external devices attached to the human body. The objective is to obtain a continuous representation of the user's environment. At present, wearable cameras are offering new ways to increase human-computer interactions mainly by allowing the user to move freely and view any given scene without being constrained by fixed cameras. The Pointer approach is based on the idea *where I am looking is what I want*. Currently, the device most employed to obtain a user's gaze is the eye-tracker. The eye-tracker allows the tracking of eyes movements by producing an estimated position of the user's gaze in real-time relative to an image frame.

Discussion: In general, most methods to detect human motion have been designed to employ passive and wearable approaches. These methods have proven to be effective in representing the action that takes place in the scene [13], [14]; unfortunately, they cannot interpret user intention (defined as the reach-to-grasp action towards an object) because they are not designed to capture the user's visual system. For example, the system proposed by Sakita, et al. [5] exploits human gaze to support cooperative work with robots. Another example was proposed by Perini, et al. [15] with the purpose of increasing user interaction for disabled people to overcome motorial difficulties. The main restrictions of these methods stem from the fact that eye-trackers are not designed to analyze hand trajectory toward an object, because the eye-tracker has no vision of the arm or hand. To overcome these drawbacks, this paper investigates a method that combines the wearable and pointer paradigm approaches.

III. Proposed method

As previously stated, our approach consists of detecting reach-to-grasp movements before the user reaches the desired object. Firstly, we use only the information provided by the camera beneath the user's wrist. The main idea is to detect reach-to-grasp movements using an HMM framework. Secondly, the prediction of gesture recognition uses a second HMM that combines reach-to-grasp movements with an eye-tracker. When a user wants to grasp an object, its gaze and hand trajectory remain almost stable for a short time. This information is used later to differentiate between fixations and grasping intentions. In the following, we describe our approach to detect: A) Hand motion recognition, and B) Grasp intention recognition based on previous results.

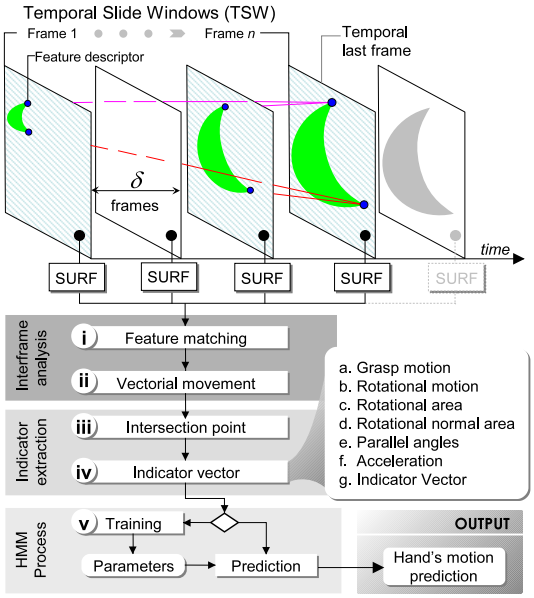


Fig. 2. Proposed hand intention recognition model based on the analysis of temporal slide windows (TSW) inter spaced by δ-frames

A. Hand motion recognition

Suppose that a user is performing a reach-to-grasp movement towards an object, one can infer that the trajectory remains steady. Accordingly, all other objects start to disappear from the user's scene. Conversely, if the hand movement is too stochastic, the probability that a user is performing a reach-to-grasp movement is reduced because the motion-descriptor does not have an approaching pattern. The main problem facing researchers is how to build a robust pattern of motion. In our problem, the number of corresponding points detected in time is low and their distribution is not uniform within the image. Likewise, the distance between objects and the camera is very close and movement is very fast; at least at 50 cm/s. In these conditions, classic methods for camera movement estimation are not well suited because they suppose an *a priori* scene model or a large set of resilient points. For these reasons our method is based on several cues related to observed motions and extracted from a robust pattern of motion based on Temporal Slide Windows (TSW). These cues are then applied to a HMM framework in order to recognize four normal hand motions, namely zoom-in, zoom-out, translation and rotation. In natural prehensive gestures (without obstacles), these movements are not completely mixed. A general overview of hand motion recognition is presented in Fig.2.

Movement representation using a TSW approach: The main idea is to relate multiple corresponding points in order to estimate global motion features or indicators[2] in each TSW, which corresponds to i) $\mapsto$ iv) steps in figure 2.

i) Feature Matching: Firstly, we extract invariant interest-

[2]in order to reduce ambiguities between motion features (or grasp features in section III-B) and SURF features, we will call the motion features indicators.

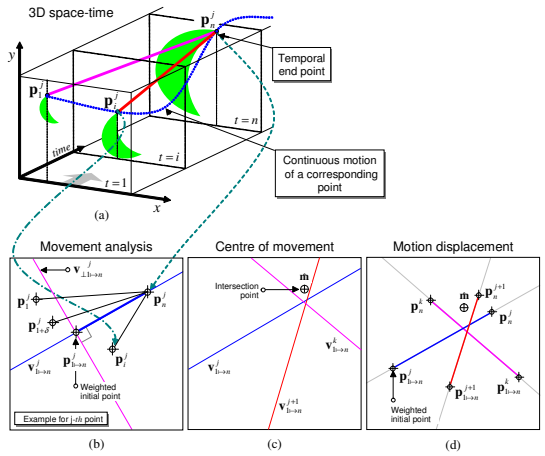


Fig. 3. Schematic view of point correspondence in time-space. (a) Corresponding points in 3D time-space volume; (b) Corresponding points in 2D coordinates.

points by means of the SURF algorithm each δ-frame contained in one TSW. This is schematically outlined in Fig.3. For instance, let $\mathbf{p}_1^j = [x_1^j, y_1^j, 1]^\top$ be the *j-th* interest point position at time $t = 1$, and $\mathbf{p}_n^j$ the interest point at time $t = n$ (last frame of the TSW). In order to seek a corresponding relationship between both interest points we use the NNDR criterion [16] between each $\mathbf{p}_1^j$ point and $\mathbf{p}_n^j$ point.

ii) Vectorial movement: By applying the same procedure to other images of the same TWS, we can build a global map vector that converges to the point $\mathbf{p}_n^j$. In order to establish a motion field along this time, several vectors of the same point are required. Namely, let $\mathbf{q}_{i,n}^j$ with $i \in \{1, \ldots, n-\delta\}$ be a homogeneous vector[3] that relates points $\mathbf{p}_i^j \mapsto \mathbf{p}_n^j$. For this, we define the general motion of multiple vectors that arrive at the point $\mathbf{p}_n^j$ as $\mathbf{Q}_{1\mapsto n}^j = \left[\mathbf{q}_1^j, \ldots, \mathbf{q}_i^j, \ldots, \mathbf{q}_{n-\delta}^j\right]^\top$ Matrix $\mathbf{Q}_{1\mapsto n}^j$ defines the motion field for point *j-th* for all frames until time $t = n$ (for each δ-frames). Nevertheless, this procedure does not assure that in every δ-frames there is correspondence because of high geometric and photometric distortions, or partial occlusions that could be present in some frames. To assure that the motion field is correct, we define a parameter ρ as the minimum number of rows in the matrix $\mathbf{Q}_{1\mapsto n}^j$ where $inliers \geq \rho$ is fulfilled. However, if this last constraint is not fulfilled, we discard the motion field for that point, in order to assure that the linear system is correct. The next step is to derive only one vector that represents the motion of point *j-th* along time. For this, we map the angle of the *j-th* feature point along all *inliers*-frames as $\mathbf{F}_{1\mapsto n}^j = \left[\mathbf{F}_{1,n}^j, \ldots, \mathbf{F}_{i,n}^j, \ldots, \mathbf{F}_{n-\delta,n}^j\right]$, where $\mathbf{F}_{1\mapsto n}^j$ is a $(1 \times inlier)$ angle vector of the SURF feature vector extracted for each δ-frames for point *j-th*. In other words, each angle $\mathbf{F}_{i,n}^j$ weighs the relative significance between the features of points $\mathbf{p}_i^j \mapsto \mathbf{p}_n^j$. Thus, the smaller the angle between two vectors, the stronger the relation of the same point. Conversely, when the angle-value is maximal, it could be considered as noise. Based on such observation, we propose to represent each angle-value as a weight vector after a linear transformation. Hence, the vector $\mathbf{F}_{1\mapsto n}^j$ is transformed to vector $\tilde{\mathbf{F}}_{1\mapsto n}^j$, used for weighing each motion vector such that

$$\tilde{\mathbf{F}}_{1\mapsto n}^j = 1 - \frac{\alpha \mathbf{F}_{1\mapsto n}^j}{\max\left(\mathbf{F}_{1\mapsto n}^j\right)}. \tag{1}$$

Experimentally α was fixed at 0.98 to use all vectors mapped in $\mathbf{F}_{1\mapsto n}^j$. Nonetheless, the vector $\tilde{\mathbf{F}}_{1\mapsto n}^j$ is not correctly scaled. To determine a correct scale measure, we compute $\mathbf{N}_{1\mapsto n}^j$ as

$$\mathbf{N}_{1\mapsto n}^j = \frac{\tilde{\mathbf{F}}_{1\mapsto n}^j}{\sum_{i=1}^{inlier} \tilde{\mathbf{F}}_{1\mapsto n}^j(i)}, \tag{2}$$

where $\sum_{i=1}^{inlier} \mathbf{N}_{1\mapsto n}^j(i) = 1$. The resultant vector $\mathbf{N}_{1\mapsto n}^j$ gives a correct measure of each angle value by taking into account the relative significance between the angles contained in $\mathbf{F}_{1\mapsto n}^j$. Finally, we compute the global vector of point *j-th* as the vector

$$\mathbf{v}_{1\mapsto n}^j = \mathbf{Q}_{1\mapsto n}^{j\top} \mathbf{N}_{1\mapsto n}^{j\top}, \tag{3}$$

where $\mathbf{v}_{1\mapsto n}^j$ is a (1×3) that maps all $\mathbf{Q}_{1\mapsto n}^j(k)$ vectors into a single one by giving more value to vectors with more similarity (straight line in Fig.3b-c). Additionally, we compute the normal directional vector in order to detect rotational movements. For this, we define matrix $\mathbf{Q}_{\perp}{}_{1\mapsto n}^{\ j}$ as the normal motion field for point *j-th*. Therefore the normal global vector is $\mathbf{v}_{\perp}{}_{1\mapsto n}^{\ j} = \mathbf{Q}_{\perp}{}_{1\mapsto n}^{j\top} \mathbf{N}_{1\mapsto n}^{j\top}$ (Fig.3b).

iii) Intersection point: We now turn to the problem of estimating the intersection point of multiple corresponding points. Suppose we have determined multiple vectors $\mathbf{v}_{1\mapsto n}^{\Theta}$, where $\Theta = \{1, \ldots, j, \ldots, k\}$ is the set of interest points detected between time $t = 1$ and $t = n$ and k is the last point in correspondence. For this, let $\mathbf{A}_{1\mapsto n}^{\Theta}$ be a $(k \times 3)$ matrix that encodes all motion vectors as $\mathbf{A}_{1\mapsto n}^{\Theta} = \left[\mathbf{v}_{1\mapsto n}^1, \ldots, \mathbf{v}_{1\mapsto n}^j, \ldots, \mathbf{v}_{1\mapsto n}^k\right]^\top$.

Experimentally, when a reach-to-grasp movement has been initiated, multiple vectors intersect a common point, called *intersection point*. To estimate the position of the unknown intersection point, we formulate a non homogeneous system of linear equations, described as follows

$$\underbrace{\begin{bmatrix} \mathbf{A}_{1\mapsto n}^{\Theta} \\ 0 \quad 0 \quad 1 \end{bmatrix}}_{\mathbf{H}} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{0}_{\mathbf{k}\times\mathbf{1}} \\ 1 \end{bmatrix}}_{\mathbf{b}}. \tag{4}$$

Changing the notation in matrix terms, (4) can be expressed as $\mathbf{Hm} = \mathbf{b}$, where $\mathbf{H}$ is an over determined matrix coefficients of $\mathbf{A}_{1\mapsto n}^{\Theta}$ vectors. To resolve this problem we use the **QR** transformation [17]. Therefore, the solution for the non homogeneous system, using the **QR** transformation is $\hat{\mathbf{m}} = \mathbf{R}^{-1}(\mathbf{Q}^\top \mathbf{b})$. In the same way, there is an *normal intersection point* defined as $\hat{\mathbf{m}}_\perp$ that represents the intersection of vectors $v_{\perp}{}_{1\mapsto n}^{\Theta}$.

[3]The vector $\mathbf{q}_{i,n}^j$ is established between time $t = i$ and $t = n$ only for the *j-th* point assuming correct matching. $\mathbf{q}_{i,n}^j$ is defined as $\mathbf{q}_{i,n}^j = \mathbf{p}_i^j \times \mathbf{p}_n^j = [x_i^j, y_i^j, 1] \times [x_n^j, y_n^j, 1]$.

iv) Indicators extracted: Below is an explanation of the eight motion features or indicators proposed to predict different hand movements.

a. Grasp motion: The first two indicators proposed are related to reach-to-grasp movements. In general, reach-to-grasp motion can be split up into two different events. *Zoom-in*: when the hand is moving towards an object; and *Zoom-out*: when the hand is moving away from an object. Whichever movement is performed, there will be an intersection point $\hat{\mathbf{m}}$ contained in the TSW. Here, we propose a simple procedure to infer whether a hand is reaching for an object or not. Firstly, let $\mathbf{P}^j_{1\mapsto n}$ be a $(inliers \times 3)$ matrix representing the 2D position in time $[1,\dots,n]$ for each δ-frames (Fig.3c): $\mathbf{P}^j_{1\mapsto n} = \left[\mathbf{p}^j_1,\dots,\mathbf{p}^j_i,\dots,\mathbf{p}^j_n\right]^\top$.

Then, we re-map the motion field by taking into account the scale matrix $\mathbf{N}^{j\top}_{1\mapsto n}$. We define $\mathbf{p}^j_{1\mapsto n}$ as a weighted mean position[4] of vector $\mathbf{v}^j_{1\mapsto n}$. Extending this procedure for all Θ-points, let $\mathbf{p}^\Theta_{1\mapsto n}$ be the motion of each point in the TSW in $[1,\dots,n]$, and let $\mathbf{p}^\Theta_n$ be the final position of each point, defined as, $\mathbf{p}^\Theta_{1\mapsto n} = \left[\mathbf{p}^1_{1\mapsto n},\dots,\mathbf{p}^k_{1\mapsto n}\right]^\top$ and $\mathbf{p}^\Theta_n = \left[\mathbf{p}^1_n,\dots,\mathbf{p}^k_n\right]^\top$. Since vector $\mathbf{p}^\Theta_{1\mapsto n}$ codes the initial weighted position, let $d_{1,m}$ be the Euclidean distance of each vector $\mathbf{p}^\Theta_{1\mapsto n}$ in relation with the intersection point $\hat{\mathbf{m}}$, and let $d_{n,m}$ be the Euclidean distance of each final position $\mathbf{p}^\Theta_n$ in relation with same intersection point $\hat{\mathbf{m}}$ as $d_{1,m}(j) = \|\mathbf{p}^\Theta_{1\mapsto n}(j) - \hat{\mathbf{m}}\|$ and $d_{n,m}(j) = \|\mathbf{p}^\Theta_n(j) - \hat{\mathbf{m}}\|$. Since we know the estimated position of the initial, final, and intersection points, the next step is to determine whether the movement is Zoom-in or Zoom-out. Based on these values, we define a function $v(j)$, as the number of nearest points to the intersection point as follows,

$$v(j) = \begin{cases} 1 & \text{if } d_{n,m}(j) \geq d_{1,m}(j) \\ 0 & \text{otherwise.} \end{cases}$$

Using the resultant function value we define two parameters (g_1, g_2) which are the mean $g_1 = \mu(v)$ and the variance $g_2 = \sigma^2(v)$. Indeed, $g_1 \mapsto 1$ when movement is Zoom-in (and conversely, $g_1 \mapsto 0$ when movement is Zoom-out). To confirm this prediction, variance (σ^2) should be low in any case.

b. Rotational motion: The rotational motion indicator gives a temporal variation of each point in correspondence. The main idea is to capture rotational movements independently of its turn direction, and thus, to compute the angle velocity of each point. Firstly, suppose that the link between $\mathbf{p}^j_i \mapsto \mathbf{p}^j_n$ and $\mathbf{p}^j_\lambda \mapsto \mathbf{p}^j_n$ exists. Therefore, s^j_i and s^j_λ are two consecutive slopes of point *j-th* separated by λ-frames respectively, defined as $s^j_i = (y^j_i - y^j_n)/(x^j_i - x^j_n)$ and $s^j_\lambda = (y^j_\lambda - y^j_n)/(x^j_\lambda - x^j_n)$.

Since both points are aiming at the last point $\mathbf{p}^j_n$ in time $t = n$, transitivity also implies that $\mathbf{p}^j_i \mapsto \mathbf{p}^j_\lambda$, where $t_\lambda > t_i$. Thereby, the angle between these consecutive slopes is $\theta^j_{i,\lambda} = \arctan\left|(s^j_i - s^j_\lambda)/(1 + s^j_i s^j_\lambda)\right|$,. Based on this result, we calculate the angular velocity ω between $\mathbf{p}^j_i$ and $\mathbf{p}^j_\lambda$ so as to compute the motion variation along time, defined as $\omega^j_{i,\lambda} = (\triangle\theta^j_{i,\lambda})/(\triangle t_{i,\lambda})$, for all $i \in \{1,\dots,inliers\}$, where $\triangle t_{i,\lambda}$ is the time difference between two consecutive frames. Combining the above value with the Euclidean distance between points $\mathbf{p}^j_i$ and $\mathbf{p}^j_\lambda$ we propose the third indicator as follows:

$$g_3 = \frac{\sum_{j=1}^{k}\sum_{i=1}^{inlier}\sigma^2(\omega^j_{i,\lambda})}{\sum_{j=1}^{k}\sum_{i=1}^{inlier}\sigma^2(\|\mathbf{p}^j_i - \mathbf{p}^j_\lambda\|)}, \tag{5}$$

The above indicator is able to distinguish rotational and translational movement. In the first case $g_3 > 1$ and in the second $g_3 \mapsto 0$.

c. Rotational area: The rotation area is formed by the triangle composed of the intersection point $\hat{\mathbf{m}}$, the weighted mean position $\mathbf{p}^j_{1\mapsto n}$ and the final end position $\mathbf{p}^j_n$ for each *j*-point. This indicator allows us to estimate whether the motion is going towards an object or not. The area variation of multiple points along the time-window is as follows

$$g_4 = \frac{1}{2k}\sum_{j=1}^{k} d_{1,m}(j)d_{n,m}(j)\sin(\phi^j_{1,n}) \tag{6}$$

where $\phi^j_{1\mapsto n}$ is the angle centered at $\hat{\mathbf{m}}$ and $d_{1,m}(j)$ and $d_{n,m}(j)$ are the adjacent segments.

d. Rotational normal area: When movement is purely rotational, we propose a similar indicator as in the above case; however, here we use the normal intersection point $\hat{\mathbf{m}}_\perp$ defined as follows.

$$g_5 = \frac{1}{2k}\sum_{j=1}^{k} d_{1,m_\perp}(j)d_{n,m_\perp}(j)\sin(\rho^j_{1,n}) \tag{7}$$

where $\rho^j_{1,n}$ is the angle centered at $\hat{\mathbf{m}}_\perp$. The above value is high when motion is not rotational because the intersection of normal vectors does not exist. However, when motion starts to be rotational there is a point $\hat{\mathbf{m}}_\perp$ that intersects all normal vectors ${v_\perp}^\Theta_{1\mapsto n}$. Consequently, all points have the same spin angle and a similar variation. As a consequence of previous results, we have obtained two angle variations. Combining angles $\rho^j_{1,n}$ and $\phi^j_{1,n}$ in the following indicator

$$g_6 = \frac{\sum_{j=1}^{k}\phi^j_{1,n}}{\sum_{j=1}^{k}\rho^j_{1,n}} \tag{8}$$

it allows us to obtain a variation of motion over time. For rotational movements g_6 tends to be constant. For translation movements, g_6 tends to be high and for Zoom-in and Zoom-out movements it increases or decreases respectively.

e. Parallel angles: Parallel angles gives the relative variation between the angles of each weighted mean position and its final end position. The key point of this indicator is to detect only translational movements, independently of its angle direction and orientation of the movement. The seventh indicator is defined as follows

$$g_7 = \frac{\sum_{j=1}^{k}(\|p^j_{1\mapsto n} - p^j_n\|)}{k\sigma^2(\psi)} \tag{9}$$

[4]estimated as $\mathbf{p}^j_{1\mapsto n} = \mathbf{P}^{j\top}_{1\mapsto n}\mathbf{N}^{j\top}_{1\mapsto n}$

where ψ is the angle of the absolute vector $\overrightarrow{p^j_{1\mapsto n} p^j_n}$. In general $g_7 \mapsto 0$ when the movement is rotational, and $g_7 \mapsto \infty$ when the movement is purely translational.

f. Acceleration: As mentioned before, human gestures are composed of continuous acceleration and deceleration phases. The proposed indicator, designed to detect these variations, is as follows:

$$g_8 = \frac{\sigma^2(a_x)}{\sigma^2(a_x) + \sigma^2(a_y)}, \tag{10}$$

where a^j_x and a^j_y are temporal accelerations with respect to point p^j_n by taking into account the temporal difference $t_{i,\lambda}$ for each i-frame contained in each TSW.

g. Indicator vector: In the previous steps we have proposed eight indicators that encode different motion features for each TSW. These cues are grouped into one vector $\mathbf{o}_1 \equiv \mathbf{o}_{1\mapsto n} = [g_1, g_2, g_3, g_4, g_5, g_6, g_7, g_8]^\top$. This vector is used as an input for an HMM framework. Nevertheless, in order to infer an intention, it is necessary to obtain multiple TSWs. Namely, a sequence is represented by multiple TSWs, each one composed of eight indicators $\mathbf{O} = [\mathbf{o}_1, \mathbf{o}_2, \ldots, \mathbf{o}_T]$, where $T+1$ is the total frame number contained in a video sequence and $\mathbf{O}$ is the observed symbol sequence.

Training HMM for recognition: Over the past few years, there has been a growing interest in employing HMMs in applications with spatial temporal variabilities, as for example, action or gesture recognition [11], [12]. More formally, HMM is composed of a number of N-states $\{S_1, S_2, \ldots, S_N\}$ connected by transitions, where each transition has an associated probability, defined by matrix A; an emission distribution probability, or the probability of emitting an observation in any given state, defined by matrix B; and an initial state distribution $\pi = \{\pi_i\}$. A HMM is fully specified by the triplet $\Lambda = (A, B, \pi)$. Based on the above parameters, the problem is to classify each class defined as a particular user's intention. Firstly, we create an HMM for each category using the well known Forward-Backward algorithm [18] to find the best parameters for each HMM. Once we have established the HMM parameters, our goal is to recognize an observed symbol sequence as a particular class, user's intention.

B. Grasp intention recognition

This section describes how user gaze and hand motion improves the detection of user intention. Below, we describe the general process using an eye-tracker and a camera attached to the user's wrist (Fig.4).

Grasp features: This process utilizes grasp features or new indicators combined with previous results in a new HMM framework.

Saccade detection: Many studies have shown that fixations are stable before the user initiates a grasp movement [19]. Conversely, saccade movements are not stable in any same position. Since an eye-tracker provides the (x, y) position of the eye's gaze, we compute the velocity rate $v_x(i)$ and $v_y(i)$ of a TWS for all $i = 1, \ldots, n$. Based on the above information we propose the following feature to quantify the global velocity as $h_1 = \sigma(v_x) + \sigma(v_y)$. Normally this feature has low values when the user is fixating and high values for saccade movements.

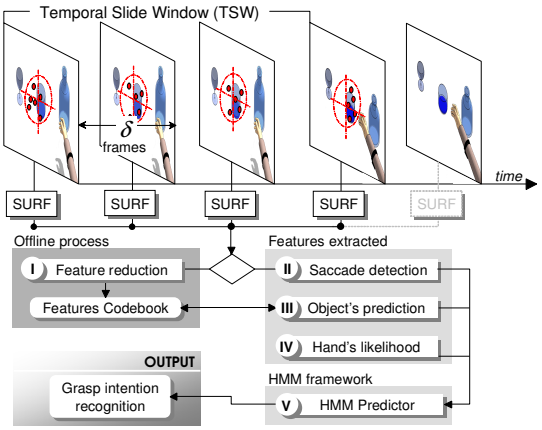


Fig. 4. Proposed user intention recognition model

Features reduction: The objective of this task is to find more resilient features over time in order to recognize the desired object in a video sequence. Here we use a similar method proposed by Sivic and Zisserman [20] to build a visual vocabulary. The key idea is that few descriptors can be seen many times throughout the video-sequence. Accordingly, more resilient features are used later to classify an object in a new video sequence. In general, there are many ways to create a codebook [21]. Here a simple method is used to compute one. First, random frames are extracted from a video sequence utilizing the user's gaze; second, each feature is classified as part of an object; and third, all other feature space is explored using the Mahalanobis distance in order to create a codebook using a Vector Quantization (**VQ**) algorithm. Once extracted, the **VQ** features of each object define the matrix $\mathbf{F_n}$ as the codebook of n-objects of interest.

Object recognition: After creating a codebook for all objects contained in the user's scene, new features are extracted from another video sequence containing all previously analyzed objects. The key idea is that some features are closer to a specific object contained in the codebook. To increase the probability of correctly classifying an object, several features contained in the same TSW have been extracted. Here we use the cosine angle distance function to measure the matching between an unknown feature vector and a known feature vector contained in the codebook. Considering a function, named *class*, that provides the class of the nearest known feature in the codebook, we compute $S_j = \sum_i \delta(class(f_i, D_n), j)$[5] for all $i = 1, \ldots, p$, where f_i is an unknown feature vector extracted from the camera's video sequence (attached to the user's head); p is the number of vectors contained in a TSW, j is the number associated with a object and $\mathbf{D_n}$ is the codebook containing an array of feature vectors. $h_2 = c/S_c = max_j(S_j)$

[5] $\delta(.,.)$ denotes the Kronecker function : $\delta(i, j) = 1$ if $i = j$, $\delta(i, j) = 0$ otherwise.

denotes the recognized object.

Hand prediction: In the previous section we described a set of features to detect hand intention based on a HMM system. Normally the outcome of this process is defined by choosing the maximal class as $arg\max_i(p(\mathbf{O}|\Lambda_i))$. However, in some situations the maximal posterior probability could be incorrect when the probability ratio between multiples class is low. For this reason we use the outcome probability of each class, given by $h_{i+2} = p(\mathbf{O}|\Lambda_i)$, for $i = 1, \ldots, 4$, where $p(\mathbf{O}|\Lambda_i)$ is the probability of detecting the i action in that TSW.

HMM for recognition: In the steps above we have defined six h_i features descriptors. These features have been designed to detect grasp movements using an HMM as shown below. Information regarding hand intention, eye position and object stability is combined because for a brief time, fixations are high, the object is always the same in both the FOV and the hand motion towards that specific object. In the same way as was defined previously, we define a new feature vector contained in a TSW as $\mathbf{o}_1 \equiv \mathbf{o}_{1\mapsto n} = [h_1, h_2, h_3, h_4, h_5, h_6]^\top$, where $\mathbf{o}_1$ is defined between time $t = 1, \ldots, n$ for the first temporal slide window. Finally the observed symbol sequence is defined as $\mathbf{O} = [\mathbf{o}_1, \mathbf{o}_2, \ldots, \mathbf{o}_T]$.

IV. Experimental results

This section presents the results of two experiments carried out with i) a camera beneath the wrist and ii) an eye-tracker[6] in parallel with a camera beneath the wrist based on the proposed framework.

Experiment 1: The goal of the first experiment is to evaluate the performance of the proposed eight features in order to predict correctly each grasping movement. At this stage we have employed five video sequences at 30 fps digitized into 320x200 pixel with 256 gray-level images, from this, we have analyzed 7131 TSWs of 21544 frames using multiple objects. We have used 1466 TSWs to train an HMM, obtained from a mug without markers on the surface. To evaluate the performance, we consider that an action is correct if the motion contained on that TSW was predicted correctly. Additionally, the system must be independent of the objects contained in the scene. In general, the performance of an HMM varies according to the data used for testing. Therefore, in our experiments we used the cross-validation method with $k = 10$. Here, we aim to evaluate the performance in videos with other objects, for this reason we tested each HMM on five different objects performing each particular action with only one object at time. Namely a cup, bottle, mug, box, and a stick of deodorant.

Figure 6 shows the average performance of ten HMMs using five different objects. We observe an average performance[7] of F-score= 0.85. In relation to reach-to-grasp movements, the Zoom-In action had the lowest performance because it is normally classified incorrectly as a rotary movement. On the contrary, Zoom-out action had, on average, the best performance near to 90%. Figure 5a and 5b reveals that performance varies according to the object being analyzed. In relation with the object performance, the bottle had the lowest performance because the SURF algorithm was unable to detect a large number of descriptors. Therefore, fewer descriptors are not conducive to building a robust TSW. On the other hand, the mug had the best performance because a large number of descriptors is used to build robust features, as the performance shows in Fig.5b.

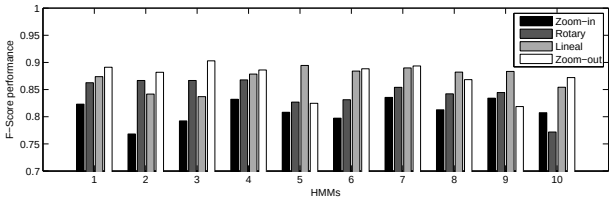


Fig. 6. Average performance of the F-Score using five objects

In our experiments we used the best HMM generated with the cross validation method. For this task we selected the best performance of each action, using as criterion the best F-Score and the best True Positive (TP) rate. The results show that we can increase the performance by 2% when using the best combination of HMMs with the F-Score and over 4% with the best combination of TP, as shown in Fig.5b-c.

Experiment 2: In the second experiment we aim at combining user intention with user gaze position, as we described in section III-B. Here we used four objects[8] placed on a uniform table and separated approximately by 15cm (without obstacles). Using the same configuration described in Fig.1, a user performs reach-to-grasp movements without grasping the object, and then he/she performs the same action with other objects.

To evaluate the performance, we created a synchronized video using two cameras; namely, the eye-tracker camera and the user's wrist camera. This stage was composed of 404 TSWs. Using this sequence, we evaluate the performance of the HMM to predict correctly each TSW as a fixation or reach-to-grasp movement. Although there are multiple objects on the table, here we have used an HMM trained with only one object (from the Exp. 1). The main idea is to predict hand movement actions independently of the objects contained in the scene.

Table I shows the performance obtained from this experiment as a confusion matrix; classified as True Positive Rate (TPR) and False Positive Rate (FPR). We can see that detection of reach-to-grasp movements yields a high

[6]We employed an ASL Eye-Trac 6.

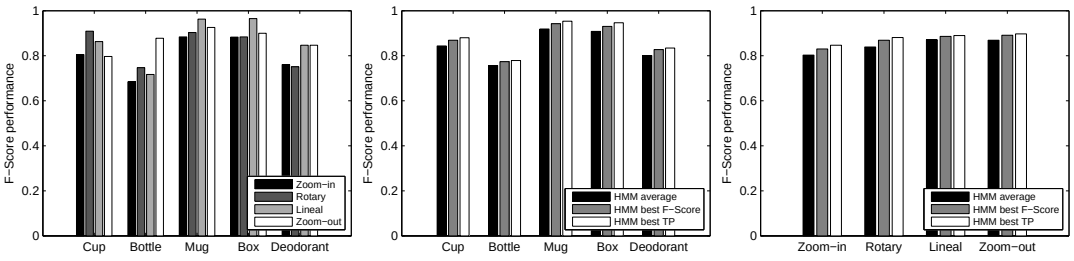
[7]F-score= 2(precision* recall)/(precision+recall)

[8]Namely a mug, a key-ring, an ID card, and a mint-box

TABLE I

Performance of the grasp intention

Class	Classified as (TSW)		Performance	
	Fixation	Reach-to-grasp	**TPR**	**FPR**
Fixation	270	54	83.3%	1.9%
Reach-to-grasp	6	74	92.5%	16.7%



(a) (b) (c)

Fig. 5. (a) Average performance for each action using all HMMs; (b) Average performance for all actions on each object using three HMM parameters; (c) Average performance for all objects on each action using three HMM parameters

TABLE II

PERFORMANCE OF THE OBJECT RECOGNITION

	Classified as				Performance	
Class	Mug	Keys	Box	Card	**TPR**	**FPR**
Mug	119	1	2	1	96.7%	0.7%
Keys	1	128	3	4	94.1%	2.2%
Box	0	2	74	1	96.1%	1.5%
Card	1	3	0	64	94.1%	1.8%
Mean					95.3%	1.6%

performance, nonetheless, there is a high false positive rate. In most of these cases, an abnormal user's behavior was observed: for example a hand coming in a good position (towards the object), but the user does not immediately initiate the gesture. This unpredicted hesitation is probably due to user's stress and desire to do the "right thing". In future investigations, we will take this into account by integrating a static hand movement estimation. Table II shows object recognition performance. We can see that the codebook constructed by the **VQ** method can be efficient, resilient and therefore appropriate when building a robust object dictionary from each TSW. It is clear that limited object numbers have been a key factor in achieving this high performance. In the future we are very interested in testing our system with more objects.

V. CONCLUSIONS

In the above experiments we show that our method can predict user grasp intention, as well as the desired object within the scene, by fusing two channels of information. The main contribution of this work for a robotic manipulator lies in our clever choice to utilize human vision combined with an active vision (micro camera placed on the user's wrist). Indeed, the Temporal Slide Window (TWS) paradigm has shown to be an efficient way to recognize human gestures and objects. It allows us to describe complex and diverse temporal visual descriptors compared to classic frame-to-frame analysis. In general, we have obtained a gesture performance between 80% and 90%. Although the objects analyzed were limited in our experiments, these results are very promising because due to the use of a limited number of resilient features. This system could also be used to enhance other human-computer interaction systems (eg. [6]). In a future investigation, we aim to further exploit the redundancy in visual information provided from both points-of-view.

REFERENCES

[1] K. Kiguchi and T. Fukuda, "A 3dof exoskeleton for upper-limb motion assist-consideration of the effect of bi-articular muscles," in *IEEE ICRA*, pp. 2424–2429d, 2004.

[2] D. Sasaki, T. Noritsugu, T. Masahiro, and H. Yamanmoto, "Wearable power assist device for hand grasping using pneumatic artificial rubber muscle," in *IEEE ROMAN*, pp. 655–660, 2004.

[3] J. Perry, J. Rosen, and S. Burns, "Upper-limb powered exoskeleton design," *Trans. on Mechatronics*, vol. 12, no. 4, pp. 408–417, 2007.

[4] H. Krebs, N. Hogan, M. Aisen, and B. Volpe, "Robot-aided neurorehabilitation," *IEEE Trans. on rehabilitation engineering*, vol. 6, 1998.

[5] K. Sakita, K. Ogawara, S. Murakami, K. Kawamura, and K. Ikeuchi, "Flexible cooperation between human and robot by interpreting human intention from gaze information," in *IROS*, pp. 846–851, 2004.

[6] Y. Tamura, M. Sugi, J. Ota, and T. Arai, "Estimation of user's intention inherent in the movements of hand and eyes for the deskwork support system," in *IEEE/RSJ IROS*, (USA), pp. 3709–3714, Nov. 2007.

[7] N. Jarrassé, J. Paik, V. Pasqui, and G. Morel, "How can human motion prediction increase transparency?," in *ICRA*, pp. 2134–2139, 2008.

[8] G. Thielman, C. Dean, and A. Gentile, "Rehabilitation of reaching after stroke: Task-related training versus progressive resistive exercise," *Arch Phys Med Rehabil*, vol. 85, p. 16131618, 2004.

[9] J. Crawford, W. Medendorp, and J. Marotta, "Spatial transformations for eye–hand coordination," *J Neurophysiol*, vol. 92, pp. 10–19, 2004.

[10] R. Johansson, G. Westling, A. Bäckström, and J. Flanagan, "Eye-hand coordination in object manipulation," *The Journal of Neuroscience*, vol. 21, no. 17, pp. 6917–6932, 2001.

[11] P. Turaga, R. Chelappa, V. Subrahmanian, and O. Udrea, "Machine recognition of human activities: A survey," *Trans. on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1473–1488, 2008.

[12] R. Poppe, "A survey on vision-based human action recognition," *Image and Vision Comp, doi:10.1016/-j.imavis.2009.11.014*, 2010.

[13] T. de Campos, W. Mayol, and D. Murray, "Directing the attention of awearable camera by pointing gestures," in *Brazilian Symposium on Computer Graphics and Image Processing*, pp. 179–186, 2006.

[14] O. Koch and S. Teller, "Body-relative navigation guidance using uncalibrated cameras," in *IEEE ICCV*, (Kyoto), 2009.

[15] E. Perini, S. Soria, A. Prati, and R. Cucchiara, "Facemouse: A human-computer interface for tetraplegic people," in *ECCV Workshop on HCI 2006*, vol. LNCS 3979, pp. 99–108, 2006.

[16] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, no. 2, pp. 91–110, 2004.

[17] N. Higham, *Accuracy and Stability of Numerical Algorithms*. SIAM, 1996.

[18] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, pp. 257–286, Feb. 1989.

[19] M. Hayhoe, A. Shrivastava, R. Mruczek, and J. Pelz, "Visual memory and motor planning in a natural task," *J Vision*, vol. 3, pp. 49–63, 2003.

[20] J. Sivic and A. Zisserman, "Efficient visual search of videos cast as text retrieval," *IEEE PAMI*, vol. 31, pp. 591–606, April 2009.

[21] A. Gersho and R. Gray, *Vector Quantization and Signal Compression*. Kluwer Academic Press, 1992.