

GENERALIZACIÓN DE LAS TRAYECTORIAS DE UN BRAZO ROBÓTICO UTILIZANDO PRIMITIVAS DE MOVIMIENTO DINÁMICO Y REGRESIÓN DE PROCESOS GAUSSIANOS

✉ CARLOS ANDRÉS PEÑA SOLÓRZANO¹
JOSÉ GABRIEL HOYOS GUTIÉRREZ²
FLAVIO AUGUSTO PRIETO ORTIZ³

RESUMEN

Es común encontrar robots realizando tareas en áreas compartidas con humanos, donde se espera que sean capaces de aprender de las acciones realizadas por otros y de adaptarse a nuevas situaciones. En este trabajo, se capturan las trayectorias del brazo de un operario mientras se mueve para agarrar un objeto, realizando seguimiento de articulaciones con el sensor kinect de Microsoft. La técnica utilizada para la codificación de las señales de entrenamiento se denominan primitivas de movimiento dinámico (DMP), mientras que la reconstrucción se realiza mediante regresión de procesos gaussianos (GPR). GPR permite además, generalizar los movimientos de entrenamiento a nuevas trayectorias, cuando cambian tanto la posición inicial de la mano como la ubicación del objeto. La técnica de generalización se compara contra un algoritmo basado en distancia de Mahalanobis y distribución gaussiana, que utiliza los datos de la trayectoria sin codificar, para realizar la estimación. La técnica propuesta presentó bajos tiempos de codificación y errores pequeños con respecto a los valores objetivo al probarlo con 30 puntos de consulta para el valor inicial de la mano, y 30 puntos para la posición final.

PALABRAS CLAVE: Robótica, aprendizaje por imitación, programación por demostración, primitivas de movimiento dinámico, regresión de procesos gaussianos.

¹ Ingeniero Electrónico. Estudiante de Maestría en Ingeniería – Automatización Industrial, Universidad Nacional de Colombia, Bogotá, Colombia.

² Ingeniero Electricista. Docente, Universidad del Quindío, Armenia, Colombia.

³ Ingeniero Electrónico. Profesor Titular, Universidad Nacional de Colombia, Bogotá, Colombia.



Autor de correspondencia: Peña Solórzano, C.A. (Carlos Andrés): Cra. 45 No. 22-44 Apt. 104A. Bogotá, Colombia.
Tel.: 6966151 / Cel.: 3013705840
Correo electrónico: capenas@unal.edu.co

Historia del artículo:

Artículo recibido: 5-V-2015/ Aprobado: 3-V-2017
Disponible online: 10 de mayo de 2018
Discusión abierta hasta octubre de 2019

GENERALIZATION OF THE TRAJECTORIES OF A ROBOTIC ARM USING DYNAMIC MOVEMENT PRIMITIVES AND GAUSSIAN PROCESS REGRESSION

ABSTRACT

It is common to find robots performing tasks in areas shared with humans, where they are expected to be able to learn from the actions taken by others and adapt to new situations. In this paper, the trajectories of the arm of an operator are captured while moving to grab an object, making joint tracking with the Microsoft kinect sensor. The technique used for the coding of the training signals is called dynamic movement primitives (DMP), while reconstruction is performed using gaussian process regression (GPR). GPR also allows generalize training movements to new paths when changing both the initial hand position and the location of the object. The technique is compared against a generalization based on Mahalanobis distance and gaussian distribution, which uses the path data to make the estimate. The proposed technique presented low encoding times and small errors regarding the target values when tested with 30 points of consultation for the initial value of the hand, and 30 points for the final position.

KEYWORDS: Robotics, learning by imitation, programming by demonstration, dynamic movement primitives, gaussian process regression.

GENERALIZAÇÃO DAS TRAJETÓRIAS DE UM BRAÇO ROBÓTICO USANDO PRIMITIVAS MOVIMENTO DINÂMICO E REGRESSÃO DOS PROCESSOS DE GAUSS

RESUMO

No percurso dos dias é comum encontrar robôs fazendo tarefas em áreas compartilhadas com os seres humanos, se deseja que eles sejam capazes de aprender com as ações realizadas por pessoas e se adaptarem a novas situações. Neste trabalho, as trajetórias do braço de um operario são capturadas enquanto se movimenta para agarrar um objeto. O seguimento das articulações é feito com um sensor Kinect da Microsoft. A técnica utilizada para codificar os sinais de treinamento é chamada de Primitivas de movimento dinâmico (DMP), enquanto a reconstrução é feita por meio de regressão processos de Gauss (GPR). A técnica GPR permite também generalizar os movimentos de treinamento para novas trajetórias quando mudam tanto a posição inicial da mão como a localização do objeto. A técnica de generalização é comparada com um algoritmo baseado na distância de Mahalanobis e a distribuição de Gauss, ela usa os dados da trajetória não codificados, para fazer a estimativa. A técnica proposta mostrou tempos de codificação baixos e erros menores em relação aos valores-alvo, quando testado com 30 pontos de consulta para o valor inicial da mão, e 30 pontos para a posição final do objeto.

PALAVRAS-CHAVE: Robótica, aprendizagem por imitação, programação por demonstração, primitivas de movimento dinâmico, regressão de processos de Gauss.

1. INTRODUCCIÓN

La investigación en áreas como la robótica y los sistemas autónomos ha presentado grandes avances en los últimos años (Lopes & Santos-Victor, 2007 y Lopes et al., 2010). Dichos avances han llevado al desarrollo de plataformas con capacidades motoras, perceptuales y cognitivas cada vez más complejas. Sumado a esto, el área de la robótica está cambiando rápidamente de ambientes industriales controlados hacia ambientes dinámicos con interacción humana, debido principalmente al incremento de robots de servicio, los cuales se espera sean parte de muchas actividades humanas en el futuro próximo (Leone et al., 2011b).

El aumento de dicha interacción presenta grandes desafíos relacionados con el aprendizaje robótico y las capacidades de interactividad (Nicolescu & Mataric, 2005). Las metodologías tradicionales de programación e interfaz con los robots no son suficientes dada la dinámica de los nuevos entornos a que están expuestos. Los autómatas requieren de la capacidad de aprender a realizar nuevas tareas de acuerdo con las preferencias de los usuarios, además deben mejorar el desempeño durante su ciclo de vida y permitir que personas no expertas puedan programar nuevas actividades robóticas de forma natural (Lopes et al., 2010). Como respuesta a estos problemas, se presentan las técnicas de aprendizaje basadas en demostración (LbD por sus siglas en inglés), llamadas también programación por demostración (PbD por sus siglas en inglés), aprendizaje por imitación (LbI por sus siglas en inglés) y aprendizaje por observación, entre otros (León et al., 2011).

En esta forma de programación, el robot, provisto de todos los sensores y hardware físicos necesarios para realizar una tarea, observa como esta es realizada por un humano u otro robot, e intenta imitar los movimientos y/o cumplir el objetivo de la tarea. Por esto, recientemente dichas técnicas de aprendizaje han atraído la atención de los investigadores en robótica, al considerarlo un mecanismo de aprendizaje prometedor para transferir conoci-

mientos de un experto a un agente artificial (Lopes & Santos-Victor, 2007).

En este trabajo se propone la programación por demostración de las trayectorias que debe seguir el brazo de un robot para alcanzar un objeto. Los datos de entrenamiento se obtienen de un operario que realiza la acción a distintos puntos objetivo en el espacio. Utilizando las funciones propias del sensor Kinect de Microsoft, es posible calcular la ubicación de las articulaciones del operario al realizar las tareas. Para que el robot pueda adaptarse a cambios en el entorno, se propone utilizar una técnica de regresión para generalizar a nuevas trayectorias dadas nuevas posiciones iniciales y finales de la mano.

La generalización de trayectorias se ha utilizado para varias aplicaciones, por ejemplo, para enseñar a un robot a atar nudos sin importar la posición inicial de la cuerda (Lee et al., 2014), para generación de letras a mano alzada a partir de muestras dadas (Kulvicius et al., 2012), y para generalizar el movimiento al lanzar dardos y al golpear en ping-pong (Kober et al., 2011). Los artículos revisados suelen enfocarse en cómo debe actuar el robot ante un cambio en la posición final o un cambio en el objetivo. En este trabajo, se presenta una técnica que permite una variación tanto en la posición inicial como final, la cual aprovecha la característica de DMP de crear trayectorias suaves, después de hallar los nuevos valores iniciales y finales para las articulaciones del robot.

Para la reproducción de los movimientos aprendidos, se plantea utilizar el simulador del robot iCub (Metta et al., 2008). El robot fue diseñado como una plataforma común para investigadores interesados en el estudio de sistemas cognitivos artificiales. El iCub es del tamaño aproximado de un niño de tres años, y está equipado con un gran número de sensores, la mayoría de los cuales pueden utilizarse durante las simulaciones. Una de las características principales del simulador, es que la programación es igual a la que se realizaría sobre el robot real.

El documento se encuentra organizado de la siguiente forma: en la Sección 2, se presentan los

trabajos relacionados en las áreas de aprendizaje por imitación y adquisición de datos. En la Sección 3, se describe el experimento realizado y la adquisición de las trayectorias de entrenamiento. En la Sección 4, se presenta la fundamentación matemática e implementación de las técnicas: primitivas de movimiento dinámico y regresión de procesos gaussianos, además de un algoritmo de generalización basado en distancia de Mahalanobis y distribución gaussiana, con el cual se compara el desempeño de la técnica empleada. En la Sección 5 se presentan la plataforma de simulación y los resultados, concluyendo en la Sección 6.

2. TRABAJOS RELACIONADOS

Las primitivas de movimiento dinámico (DMP por sus siglas en inglés), permiten codificar los movimientos del robot, donde una DMP puede codificar una trayectoria específica. En el caso de un movimiento discreto (punto a punto), cada grado de libertad del robot se describe por un sistema de ecuaciones diferenciales no lineales (Kormushev et al., 2011), lo que le permite reaccionar a perturbaciones sin introducir discontinuidades en el movimiento resultante. Además, DMP no es directamente dependiente del tiempo, por lo que es fácil detener la ejecución del movimiento sin tener en cuenta la evolución del tiempo de forma extensiva.

Las DMPs pueden ser extendidas para incluir capacidad de evitar obstáculos, cambiando de una primitiva a otra mediante retroalimentación sensorial (Forte et al., 2011) o creando posiciones objetivo virtuales utilizando campos potenciales (Tan et al., 2011). En (Nemec & Ude, 2012) se propone realizar acciones secuenciales, asegurando que las primitivas de movimiento consecutivas se unen en forma continua. Además de replicar los movimientos o acciones aprendidas, utilizando una reconstrucción probabilística basada en regresión de procesos gaussianos (GPR por sus siglas en inglés), pueden crearse acciones para situaciones que el robot no ha encontrado anteriormente, lo cual se presenta tanto en simulación como en experimentos reales de

alcanzar posiciones, tomar objetos y lanzar esferas (Ude et al., 2010 y Forte et al., 2012).

DMP se ha utilizado con aprendizaje por refuerzo, donde los objetivos y los parámetros de tiempo se fijan, y sólo se aprenden los pesos para ajustar la forma de la trayectoria. En (Tamosiunaite et al., 2011) se utilizaron técnicas de aproximación al valor de la función para el aprendizaje del objetivo, y métodos de búsqueda de política directa para predecir los pesos. Los resultados fueron comprobados al aprender a verter un líquido, tanto en simulación como en un brazo robótico. El método demostró ser estable y robusto. Por su parte, en (Stulp et al., 2011), se utilizó aprendizaje por refuerzo para enseñarle al robot primitivas de movimiento robustas ante incertidumbres en la estimación del estado. Específicamente, durante maniobras de alcance de objetos, el robot aprende estrategias de manipulación fina para llevar al objeto a una orientación en que el agarre tenga mayores posibilidades de éxito.

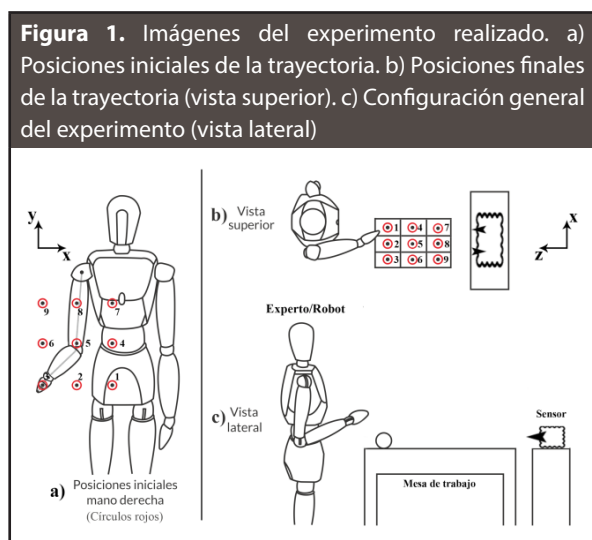
Se ha difundido de manera amplia el aprendizaje robótico por imitación utilizando imágenes como datos de entrada (León et al., 2011). En el pasado los sensores que proveían imágenes 3D eran de difícil adquisición dada su poca disponibilidad y altos precios. En Noviembre de 2010, Microsoft lanza el sensor Kinect con la capacidad de producir flujos de imágenes RGB y de profundidad a un precio mucho menor a los sensores de rango tradicionales. Entre los desarrollos utilizando dicho sensor, se encuentran el seguimiento en 3D de las articulaciones del esqueleto humano, o partes de este (Oikonomidis et al., 2011), además de captura de movimientos con los cuales determinar comportamientos e intenciones (Toda et al., 2011 y Kaneko et al., 2011).

En Pastor y otros (Pastor et al., 2009), se presenta una modificación a DMP, la cual consiste en variaciones leves de esta. La técnica permite cambios en el punto inicial y final de una sola trayectoria original, también agregan términos a DMP para la evasión de obstáculos. Su desventaja es que no generaliza con información de múltiples trayectorias.

3. ADQUISICIÓN DE LOS DATOS

3.1. Configuración del experimento

El experimento consiste en alcanzar un objeto que se encuentra sobre una mesa, captando el movimiento de la mano derecha con un sensor Kinect (**Figura 1**). Quien realiza las demostraciones está de pie frente a la mesa donde están indicadas las nueve posiciones posibles del objeto. Los nueve valores iniciales se calculan teniendo en cuenta la posición de la mano, cuando el brazo está extendido completamente junto al cuerpo, y la ubicación de la cintura. La base de datos consiste de 81 trayectorias, nueve posiciones iniciales de la mano derecha (**Figura 1a**), de las cuales se realiza el movimiento a cada uno de los nueve valores finales (**Figura 1b**). En una mesa, más adelante, se ubica el sensor kinect, con el cual se obtienen las imágenes 2D y 3D que permiten utilizar el algoritmo para la ubicación de los puntos (x ; y ; z) de las articulaciones.



A diferencia de otros proyectos donde se generaliza con GPR a nuevas posiciones finales (Forte et al., 2011 y Forte et al., 2012), en este trabajo se generaliza ante variaciones tanto en el punto inicial como final. Se utiliza GPR para predecir los valores de los ángulos iniciales de acuerdo al punto de inicio (punto de consulta q_i), y GPR nuevamente para

determinar los parámetros de una nueva DMP (w_s , g_s , τ_s), que produzca una trayectoria suave entre los nuevos valores iniciales de las articulaciones, y valores que lleven la mano a la nueva posición del objeto.

Para contar con una base de datos con mayor información, se realizan ajustes a los valores iniciales y finales de los ángulos obtenidos, cubriendo una mayor área antes de llevar las trayectorias al robot (**Figura 2**).

En la **Figura 2**, las estrellas indican las posiciones iniciales (plano (X, Y)) y finales (plano (X, Z)) de los movimientos realizados por el operario en el espacio cartesiano, mientras que las figuras restantes son movimientos no realizados por el usuario, sino estimados modificando los ángulos de las 4 trayectorias obtenidas. Las posiciones iniciales y finales no se encuentran en una malla cuadrada, tal como se presenta al crear la base de datos (**Figura 1**), debido a las diferencias en longitud entre el brazo de quien realiza la acción y el brazo del robot, además de articulaciones del operario que no son tenidas en cuenta.

3.2. Ángulos de las articulaciones

Utilizando la librería NITE creada para el sensor Kinect de Microsoft, se creó un programa que reconoce el esqueleto del usuario y accede a los datos (x, y, z) de cada articulación en coordenadas del mundo; cada dato correspondiente a un eje tiene como unidad los milímetros. El valor x presenta la ubicación en el eje horizontal, perpendicular al eje de la cámara; y corresponde a la elevación del punto en forma vertical; z es la profundidad medida desde el sensor.

Las articulaciones usadas para calcular los ángulos requeridos fueron: cadera derecha e izquierda, cuello, hombro derecho, codo derecho y mano derecha (**Figura 3**). Las articulaciones del brazo izquierdo no se utilizan, ya que la demostración de la tarea consiste en llevar la mano derecha hacia el objetivo.

Figura 2. Valores extremos de las trayectorias en el espacio cartesiano

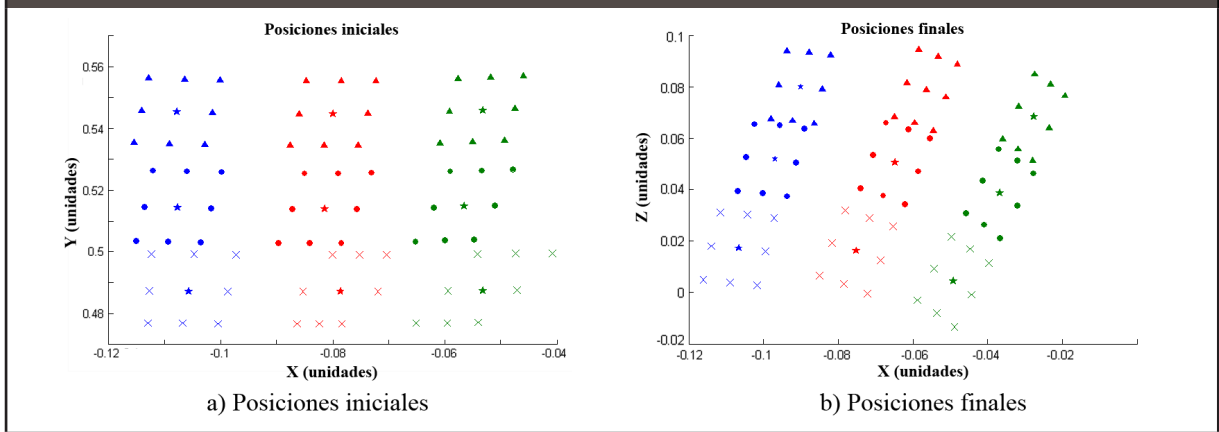
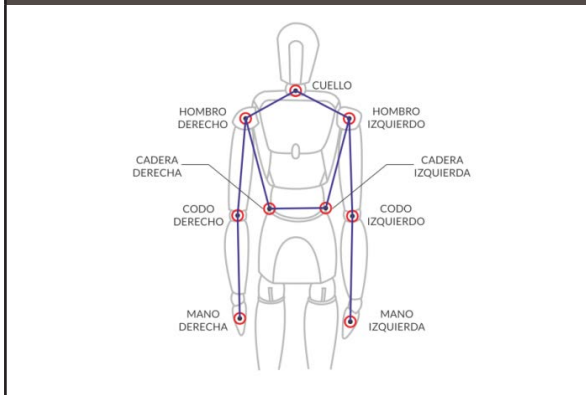
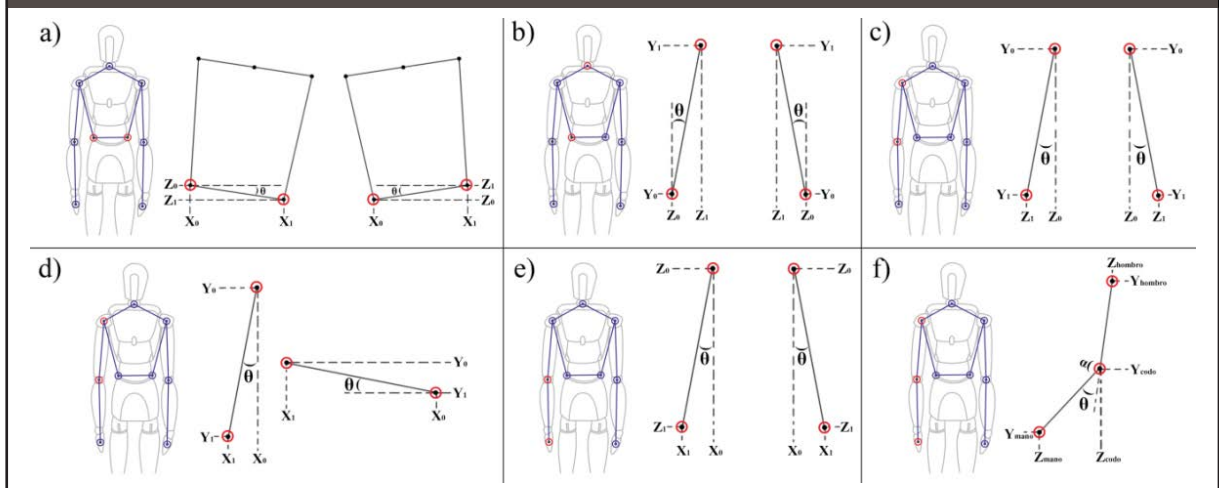


Figura 3. Articulaciones del cuerpo utilizadas para el cálculo de los ángulos



Los ángulos calculados fueron: guiñada del torso (yaw), cabeceo del torso (pitch), cabeceo del hombro, balanceo del hombro, guiñada del hombro, y codo (**Figura 4**). Estos ángulos se seleccionaron teniendo en cuenta que exista un ángulo análogo en el robot, y que tenga un papel relevante en la tarea de llevar la mano derecha a la posición objetivo. En el caso específico de los ángulos del torso, se tienen en cuenta dado que modifican el alcance final de la mano. En la **Tabla 1** se describen los ángulos y cálculos requeridos con respecto a las articulaciones de la **Figura 4**.

Figura 4. Ángulos calculados de las articulaciones. a) Guiñada del torso. b) Cabeceo del torso. c) Cabeceo del hombro. d) Balanceo del hombro. e) Guiñada del hombro f) Codo



En la **Tabla 1**, los primeros cinco ángulos se calculan utilizando la fórmula de tangente, sobre el triángulo rectángulo formado por las articulaciones relacionadas en la gura. Para el ángulo del codo, se utiliza la ley de coseno sobre el triángulo formado por las articulaciones de hombro, codo y mano.

4. PRIMITIVAS DE MOVIMIENTO DINÁMICO Y REGRESIÓN DE PROCESOS GAUSSIANOS

4.1. Primitivas de movimiento dinámico

Una primitiva de movimiento dinámico puede codificar una trayectoria específica del robot. La trayectoria de cada grado de libertad y , dado en el espacio de articulaciones o de la tarea, está descrito por el sistema de **Ecuaciones diferenciales (1)-(3)** (Ude et al., 2010).

$$\tau \dot{z} = \alpha_z (\beta_z (g - y) - z) + f(x), \quad (1)$$

$$\tau \dot{y} = z, \quad (2)$$

$$\tau \dot{x} = \alpha_x x, \quad (3)$$

donde z es una variable auxiliar y x es la variable de fase utilizada para hacer implícita la dependencia de f del tiempo (con valor inicial $x(0) = 1$). α_x , α_z y β_z son variables que deben definirse de tal forma que el sistema converja al punto de equilibrio $(z, y, x) = (0, g, 0)$. La función no lineal f contiene parámetros libres, que permiten al robot realizar un movimiento suavizado punto a punto de la posición inicial y_0 , a la configuración final g .

$$f(x) = \frac{\sum_{k=1}^N w_k \psi_k(x)}{\sum_{k=1}^N \psi_k(x)} x, \quad (4)$$

$$\psi_k(x) = \exp(-h_k(x - c_k)^2), \quad (5)$$

donde ψ_k son las funciones de base radial, c_k son los centros de dichas funciones distribuidas en la trayectoria, w_k son los pesos y el ancho $h_k > 0$. w_k se estima para que el DMP codifique la trayectoria deseada. Cada grado de libertad está representado por su propio sistema de **Ecuaciones (1)-(2)**, con una fase común (3).

TABLA 1. CÁLCULO DE LOS ÁNGULOS Y RANGO DE MOVIMIENTO

Ángulo	Ecuación	Rango (°)
Guiñada del torso	$\tan\theta = \frac{z_1 - z_0}{x_1 - x_0}$ (Figura 4a)	-50, 50
Cabeceo del torso	$\tan\theta = \frac{z_1 - z_0}{y_1 - y_0}$ (Figura 4b)	-10, 70
Cabeceo del hombro	$\tan\theta = \frac{z_1 - z_0}{y_1 - y_0}$ (Figura 4c)	-95, 10
Balanceo del hombro	$\tan\theta = \frac{x_1 - x_0}{y_1 - y_0}$ (Figura 4d)	0, 90
Guiñada del hombro	$\tan\theta = \frac{x_1 - x_0}{z_1 - z_0}$ (Figura 4e)	-37, 80
Codo	$\theta = 180^\circ - \cos^{-1} \left(\frac{a^2 + b^2 - c^2}{2ab} \right) \text{ (Figura 4f)}$ $a = \sqrt{(z_{\text{hombro}} - z_{\text{codo}})^2 + (y_{\text{hombro}} - y_{\text{codo}})^2}$ $b = \sqrt{(z_{\text{codo}} - z_{\text{mano}})^2 + (y_{\text{codo}} - y_{\text{mano}})^2}$ $c = \sqrt{(z_{\text{hombro}} - z_{\text{mano}})^2 + (y_{\text{hombro}} - y_{\text{mano}})^2}$	15,5, 106

Los ajustes a los datos de entrada y la elección de los parámetros de la técnica son los que se presentan en (Peña-Solórzano, Hoyos-Gutiérrez y Prieto- Ortiz, 2015). Se comprobó el funcionamiento de la técnica, al cambiar el número de funciones de base radial N , como se presenta en la **Figura 5**. Se seleccionaron los valores $N = \{10, 15, 20\}$, ya que números más bajos presentan una reconstrucción alejada de la señal original, mientras que valores más altos no presentan variaciones considerables. Se proponen estos tres valores para determinar el cambio en el desempeño, un rango donde la fidelidad de la señal reproducida es alta.

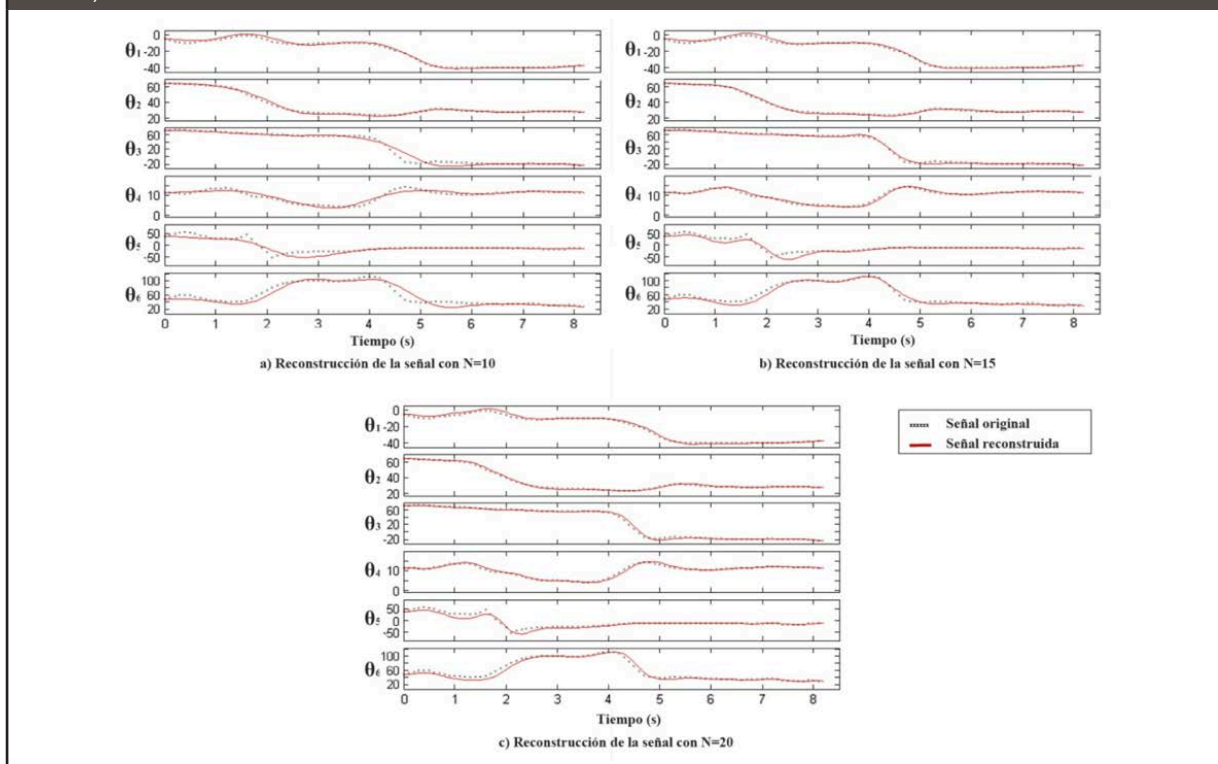
En la **Figura 5**, se observa en negro la señal original y en rojo la trayectoria reconstruida. Los errores entre ambas disminuye al aumentar el número de funciones de base radial, lo cual se nota particularmente en los puntos donde las curvas son pronunciadas, por ejemplo, a los dos segundos en los ángulos θ_5 y θ_6 .

5. REGRESIÓN DE PROCESOS GAUSSIANOS

Se propone generalizar a nuevas trayectorias a partir de otras ya conocidas, lo cual puede verse como un problema de predicción. Puede pensarse primero, que si se tienen observaciones con ruido de una variable dependiente en ciertos valores de la variable independiente x , se quiere estimar el valor de la variable dependiente en un nuevo valor x .

Si se asume cierto patrón sobre los datos de entrada, se pueden utilizar diferentes métodos para ajustar estos valores a una línea recta, o una función cuadrática, cúbica, o incluso no-polinómica. La regresión de procesos gaussianos (GPR) es una aproximación más fina, ya que no relaciona los datos a un modelo específico, sino que por medio de un proceso gaussiano se representan los valores de entrada de forma indirecta, pero rigurosa.

Figura 5. Reconstrucción de las señales con DMP. En puntos la señal original, en rojo la señal reconstruida. a) $N=10$. b) $N=15$. c) $N=20$.



Para esto, la regresión de procesos gaussianos se basa en el modelo probabilístico bayesiano (Nguyen-Tuong et al., 2009). Dado un conjunto de n datos de entrenamiento $\{x_p, y_p\}_{p=1}^n$, se desea encontrar una función $f(x_i)$ que transforma el vector de entrada x_i en el objetivo y_p , dado un modelo $y_i = f(x_i) + \epsilon_p$, donde ϵ_p es ruido gaussiano con media cero y varianza σ_n^2 . Como resultado, los objetivos observados pueden describirse por una distribución gaussiana $y \sim \mathcal{N}(0, K(X, X) + \sigma_n^2 I)$, donde X es el grupo que contiene todos los puntos de entrada x_i y $K(X, X)$ la matriz de covarianza calculada utilizando la función de covarianza dada. Una de las funciones más utilizada se basa en kernels gaussianos (**Ecuación 6**).

$$k(x_p, x_s) = \sigma_s^2 \exp\left(-\frac{1}{2}(x_i - x_s)^T W (x_i - x_s)\right), \quad (6)$$

donde σ_s^2 es la varianza de la señal y W representa el ancho del kernel gaussiano. La distribución conjunta de los valores del objetivo observado y predicho $f(x_s)$, para un punto de consulta x_s , está dada por la **Ecuación 7**.

$$\begin{bmatrix} y \\ f(x_s) \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(X, X) + \sigma_n^2 I & k(X, x_s) \\ k(x_s, X) & k(x_s, x_s) \end{bmatrix}\right). \quad (7)$$

El valor esperado $f(x_s)$, asociado al nuevo punto de consulta x_s , con su correspondiente covarianza $V(x_s)$, está dado por las **Ecuaciones 8 y 9** respectivamente.

$$f(x_s) = k_s^T (K + \sigma_n^2 I)^{-1} y = k_s^T a, \quad (8)$$

$$V(x_s) = k(x_s, x_s) - k_s^T (K + \sigma_n^2 I)^{-1} k_s, \quad (9)$$

donde $k_s = k(X, x_s)$, $K = K(X, X)$, y a denota el vector de predicción, el cual depende de los datos de entrenamiento, los datos de la variable dependiente y los valores objetivo. La parte computacionalmente más costosa es $(K + \sigma_n^2 I)^{-1}$, pero dado que esta matriz sólo depende de los datos de entrenamiento, los cálculos necesarios pueden realizarse fuera de línea. Si definimos:

$$z = (K + \sigma_n^2 I)^{-1} y. \quad (10)$$

De la **Ecuación 8**, el parámetro $f(x_s)$ puede escribirse entonces como:

$$f(x_s) = \sum_{i=1}^{NumEj} k(x_s, x_i) z_i, \quad (11)$$

donde, al ser usado para determinar los parámetros de una nueva DMP como en este caso, $f(x_s)$ se refiere a w_* , g_* y τ_* (**Ecuación 12**). Por lo tanto, los datos de entrenamiento tienen una ponderación basada en la distancia entre los puntos de entrenamiento y el punto de consulta actual. Esto resulta en que puntos de entrenamiento más cercanos tienen mayor influencia en el resultado, por lo que puede pensarse en GPR como un método de regresión local. En el **Algoritmo 1**, se presentan los pasos para utilizar la técnica regresión de procesos gaussianos.

Una de las ventajas de GPR se presenta en el ajuste de los hiperparámetros $f(x_s)$. En el **Algoritmo 1**, la técnica se entrena con los datos de entrada, y con el cálculo del logaritmo negativo de la probabilidad marginal (Rasmussen, 2004), se obtienen unos valores apropiados. Luego, se procede a realizar la predicción para cada término de la nueva DMP teniendo en cuenta el punto de consulta definido.

Algoritmo 1. Algoritmo para regresión de procesos gaussianos

- 1: Procedimiento Regresión de procesos gaussianos
 - 2: Ajuste de los datos de entrenamiento.
 - 3: Entrenamiento de GPR para ajustar los hiperparámetros.
 - 4: Predicción de los parámetros w_* , g_* y τ_* a partir del punto de consulta x_s , utilizando la **Ecuación 8**.
 - 5: Reproducción de la nueva trayectoria utilizando la DMP obtenida.
 - 6: Final del procedimiento
-

En este trabajo, se generaliza a nuevos movimientos a partir de los parámetros de pesos (w), objetivo (g) y variable de tiempo (τ) de las DMP obtenidas de las demostraciones (**Ecuación 12**) (Ude et al., 2010, Forte et al., 2011 y Forte et al., 2012).

$$G\left(\{w_i, g_i, \tau_i; q_i\}_{i=1}^{NumEj}\right): q_s \mapsto (w_*, g_*, \tau_*), \quad (11)$$

donde los parámetros w_i , g_i y τ_i son los parámetros de las DMP de entrenamiento, y están ligados a un punto q_i en la base de datos, el cual puede representar la posición inicial o final de la trayectoria en el espacio de trabajo. Cuando q_i se refiere al valor inicial del movimiento, los objetivos g_i son los valores iniciales de las articulaciones, mientras que cuando q_i es el valor final de la trayectoria, g_i se refiere a los valores finales de las articulaciones al realizar el movimiento. Por su parte, $NumEj$ es el número de ejecuciones o demostraciones que componen la base de entrenamiento, a partir de la cual se estiman los parámetros de una nueva DMP (w , g , y τ), dado un punto de consulta inicial y/o final q .

6. DISTANCIA DE MAHALANOBIS Y DISTRIBUCIÓN GAUSSIANA

Para comparar el desempeño de la técnica GPR, se realizó la generalización de los movimientos en la base de datos a nuevas trayectorias utilizando un algoritmo de asignación de pesos locales, la cual se basa en distancia de Mahalanobis y distribución gaussiana. Se calcula la distancia de las posiciones finales al punto de consulta o nueva posición final deseada, utilizando la distancia de Mahalanobis (De Maesschalck et al., 2000), la cual tiene en cuenta la correlación en los datos, ya que se calcula usando la inversa de la matriz varianza-covarianza (Σ) del conjunto de datos de interés (Ecuación 13).

$$d_m(\bar{x}, \bar{y}) = \sqrt{(\bar{x} - \bar{y})^T \Sigma^{-1} (\bar{x} - \bar{y})}. \quad (13)$$

Basado en la distancia, se asignan pesos a las trayectorias utilizando una distribución gaussiana (Ecuación 14).

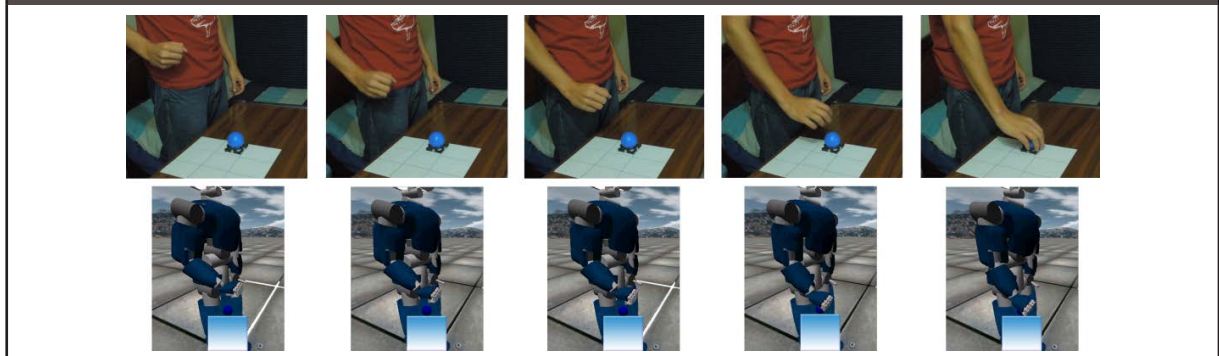
$$w = \exp(-0,5 \cdot d_m). \quad (14)$$

De acuerdo a los pesos, sólo las trayectorias por encima de cierto umbral se tienen en cuenta para la generalización. Los pesos de dichas trayectorias se normalizan a valores entre 0 y 1, y se promedian con los pesos correspondientes.

7. RESULTADOS

El robot iCub fue diseñado como una plataforma común para investigadores interesados en el estudio de sistemas cognitivos artificiales y ha sido adoptado por cerca de 20 laboratorios a nivel mundial (Metta et al., 2008). La plataforma se ha utilizado para comprobar la obtención de habilidades cognitivas para interactuar con el mundo físico que lo rodea y manipular objetos de manera flexible (Tikhonoff et al., 2011 y Natale et al., 2013). De forma más específica, se le han enseñado habilidades como la arquería (Kormushev et al., 2010) y el disparo a objetos estacionarios y en movimiento (Nath & Levinson, 2012), utilizando visión por computador y aprendizaje de máquina. Para el presente trabajo, se utiliza el simulador del robot, el cual permite la programación de la plataforma de forma idéntica a como se realiza en el robot real.

Figura 6. Secuencia de agarre de un objeto. Arriba: Acción realizada por quien realiza la demostración. Abajo: Acción realizada en el robot simulado iCub



En el entorno virtual del robot es posible crear objetos como mesas y esferas. Aquí, se utilizan las esferas para marcar las posiciones objetivo, para obtener información visual del error. A modo de ejemplo, en la **Figura 6** se muestra una secuencia del movimiento en dirección al objetivo, realizada al crear la base de datos (arriba) y al reproducirla en el robot (abajo).

Del movimiento del operario, se obtienen los valores de las articulaciones al realizar la acción, las cuales se ajustan y se codifican como DMP, para limitar el uso de espacio en memoria, y para la posterior generalización utilizando GPR.

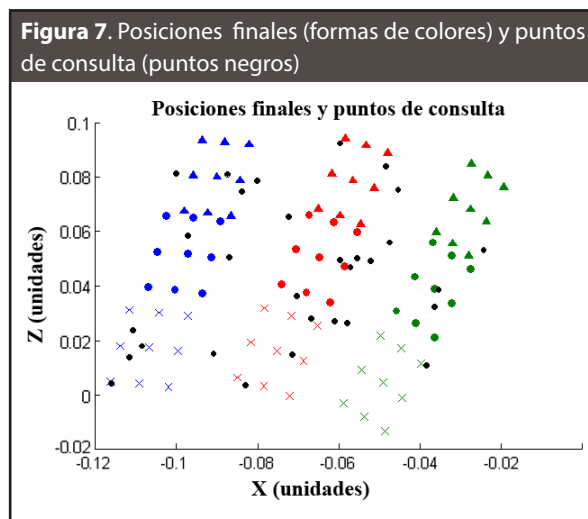
Para el cálculo del error al objetivo se utiliza la **Ecuación 15**.

$$err_{obj} = \frac{q_{x,z}^* - q'_{x,z}}{h_{x,z}}, \quad (15)$$

Donde $q_{x,z}^*$ es la coordenada del punto de consulta en x o en z , $q'_{x,z}$ es el valor final de la reproducción en el robot (en x o en z) al utilizar las técnicas GPR y distancia de Mahalanobis con distribución gaussiana, y $h_{x,z}$ es el ancho, es decir, la distancia entre los puntos más a la izquierda y a la derecha para el eje x , y más arriba y abajo para el eje z .

Para realizar la comparación entre GPR con DMP, y la aproximación utilizando la distancia de

Mahalanobis y distribución gaussiana, se crearon 30 puntos de consulta utilizando una función aleatoria uniforme, ubicados en el área entre las posiciones finales de las trayectorias (**Figura 7**).



Para un punto de consulta determinado, se obtienen los valores de pesos (w), objetivo (g) y variable de tiempo (τ) que determinan una nueva DMP (**Ecuaciones 1-3**), y por consiguiente una nueva trayectoria, para cada una de las articulaciones. En la **Figura 8**, se presenta como ejemplo uno de los puntos de consulta (**Figura 8a**) junto a las trayectorias de entrenamiento y trayectoria generalizada para uno de las articulaciones (**Figura 8b**).

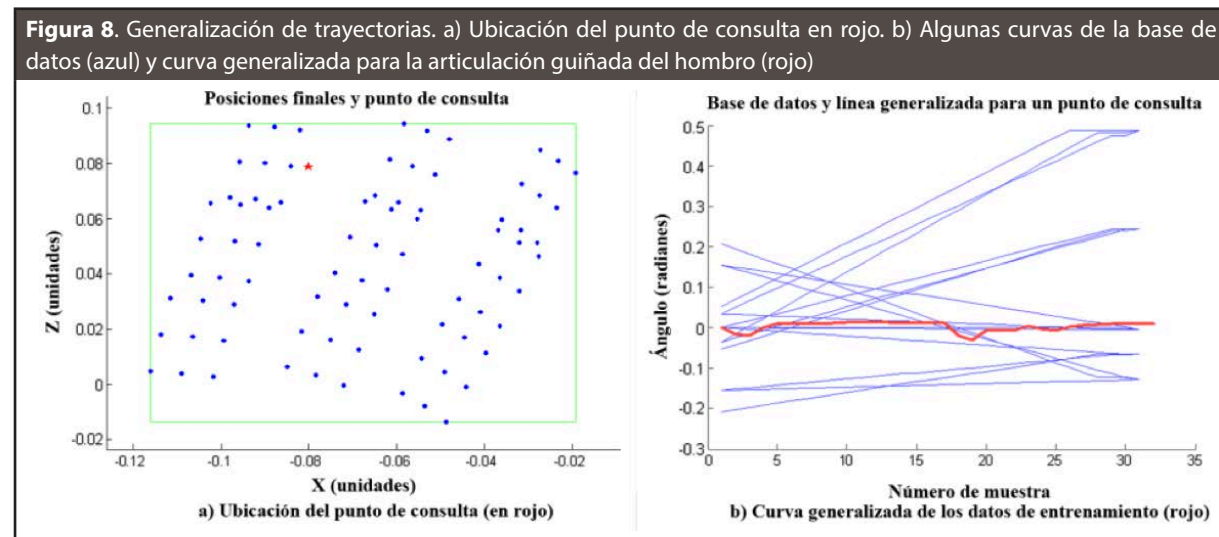
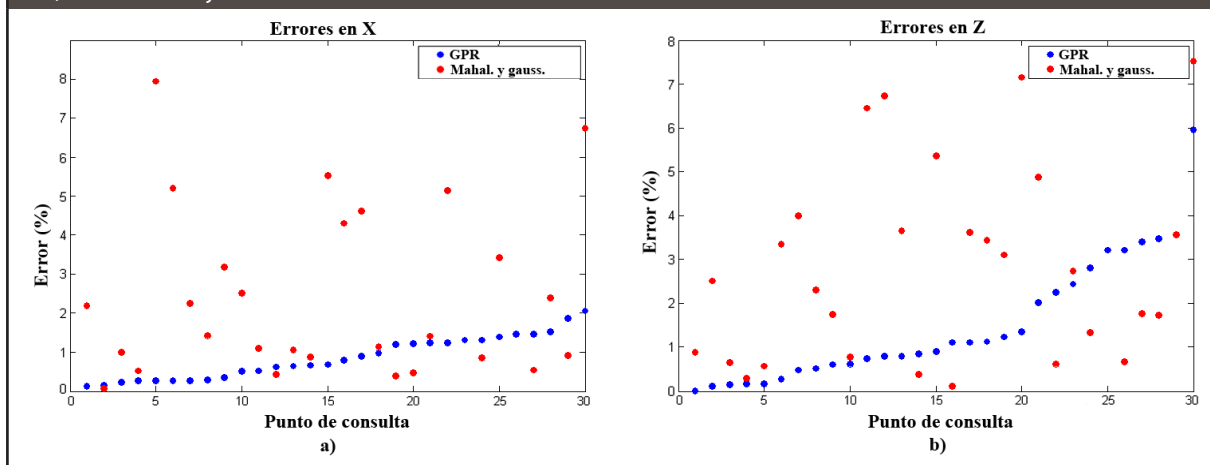


Figura 9. Errores al objetivo utilizando GPR y la distancia de Mahalanobis con distribución gaussiana. a) Errores en el eje X. b) Errores en el eje Z



Para cada una de las 30 trayectorias generadas a los nuevos objetivos, se calcula el error en los ejes X y Z al realizar el movimiento en el robot simulado iCub (**Figura 9**). En la **Tabla 2**, se presenta una síntesis de los resultados obtenidos, calculando los valores máximo, mínimo, promedio y desviación estándar.

Como se observa en la **Figura 9**, donde se organizaron los errores de menor a mayor según los valores de la técnica GPR, en algunos puntos de consulta el error obtenido con la distancia de Mahalanobis y distribución gaussiana es menor. GPR compensa esto presentando un menor error promedio, como lo presenta la **Tabla 2**. Además, el error en x presentado por GPR es cuatro veces más pequeño que el encontrado con la técnica de comparación. La desviación estándar para la regresión de procesos gaussianos se mantiene menor a 1,45% tanto en x como en z, para un valor promedio de 1,52% en z, mientras que la distancia de Mahalanobis y distribución gaussiana presenta una desviación estándar mayor a 2,3% para un promedio de 2,82% en x.

Por motivos de comparación, hasta este punto solo se han mostrado resultados de predicción de la trayectoria para alcanzar una nueva posición final. A diferencia de la técnica que se propone aquí, la técnica que utiliza la distancia de Mahalanobis y distribución gaussiana no puede realizar generalización cuando se varían tanto el punto de inicio como el

punto final, ya que las trayectorias que tienen más peso para la generalización del punto de inicio, no necesariamente serán las mismas trayectorias con mayor prioridad para generalizar el punto final.

TABLA 2. COMPARACIÓN ENTRE LA TÉCNICA EMPLEADA Y LA TÉCNICA BASADA EN DISTANCIA DE MAHALANOBIS Y DISTRIBUCIÓN GAUSSIANA

Error	DMP y GPR		Dist. Mahalanobis y dist. gaussiana	
	x	z	x	z
Máximo	2,06%	5,97%	8,78%	7,52%
Mínimo	0,12%	0,01%	0,07%	0,12%
Promedio	0,86%	1,52%	2,82%	2,98%
Desviación estándar	0,55%	1,42%	2,60%	2,33%

Para comprobar el comportamiento de la técnica GPR cuando se proponen nuevos valores iniciales y finales, se crean puntos de consulta en los planos de inicio y fin, los cuales sirven como posiciones iniciales y finales de una nueva trayectoria, la cual se estima generalizando a partir de la base de datos de trayectorias conocidas. Se crearon 30 puntos de consulta en el área de entrenamiento de las posiciones iniciales (**Figura 10**), cada uno de los cuales se emparejó con uno de los 30 puntos de consulta en el área de entrenamiento de las posiciones finales (**Figura 7**).

Tanto los puntos de consulta de posición inicial como final, se crearon utilizando una función aleatoria uniforme. En la **Tabla 3** se presenta una síntesis de los datos de error obtenidos al generalizar a un punto de consulta de inicio. De la **Tabla 3** se observa que el error promedio se mantiene por debajo del 0,8% tanto en el eje x como en el eje y , error considerado bajo.

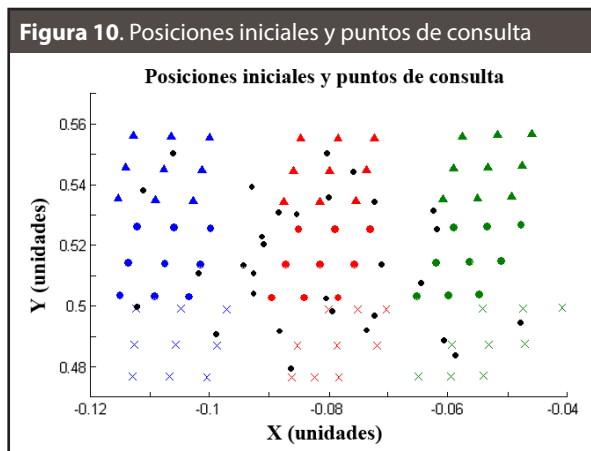


TABLA 3. DESEMPEÑO DE LA TÉCNICA EMPLEADA AL GENERALIZAR EL PUNTO DE INICIO

Error	DMP y GPR	
	X	Y
Máximo	2,68%	1,44%
Mínimo	0,08%	0,01%
Promedio	0,78%	0,72%
Desviación estándar	0,61%	0,48%

Un análisis más detallado de los errores al utilizar las técnicas GPR, y distancia de Mahalanobis con distribución gaussiana, permite observar que los máximos se producen cuando los puntos de consulta se encuentran en los extremos del área de entrenamiento. Esto se debe a que los puntos de entrenamiento cercanos sólo se encuentran por un lado del punto de consulta, por lo que no se provee a los algoritmos de generalización de datos sobre el comportamiento de las curvas en todas las direcciones. Es importante anotar que aunque se habla de los errores máximos, el error en GPR es menor al presentado en la técnica de comparación.

Los errores promedio de la técnica GPR, para generalización de punto de inicio (**Tabla 3**) y del punto final (**Tabla 2**), nunca fueron superiores al 1,6%. Este error se considera bajo, más aún cuando la desviación estándar se mantuvo menor al 1,5%. Estos resultados superan a la generalización utilizando la distancia de Mahalanobis y distribución gaussiana, que aunque en ciertos puntos presenta un error menor, el análisis estadístico de los 30 experimentos sugiere que su desempeño general es más bajo.

8. CONCLUSIONES

Se propuso la generalización a nuevas trayectorias para nuevos puntos iniciales y finales a partir de una base de datos de movimientos conocidos. Los movimientos constan de las trayectorias de los seis grados de libertad del tronco y brazo, necesarios para alcanzar un objeto ubicado en una mesa frente al operario o robot.

Para codificar las trayectorias y crear la base de datos, se utilizó la técnica de aprendizaje por imitación primitivas de movimiento dinámico. Para la generalización, se presentó la técnica regresión de procesos gaussianos, capaz de estimar los parámetros de una nueva DMP a partir de la base de datos de DMP creada con las trayectorias de entrenamiento. Para evaluar el desempeño del algoritmo presentado, se implementó una técnica basada en distancia de Mahalanobis y distribución gaussiana, la cual utiliza las trayectorias no codificadas para estimar un nuevo movimiento, dado un nuevo valor final en el espacio de trabajo del robot. El desempeño de la técnica DMP y GPR presenta un mejor desempeño, con un error menor al 1,6%, frente a un error por encima de 2,8% para la técnica de comparación.

La reproducción de las trayectorias de entrenamiento y generalizadas, se realizó en la plataforma del robot simulado iCub. La programación se lleva a cabo de forma idéntica a como se realizaría sobre el robot real. Se obtuvieron las trayectorias estimadas al crear 30 puntos de consulta entre los valores finales de los movimientos conocidos. Los resultados estadísticos obtenidos señalan un mayor

desempeño por parte de la técnica GPR con DMP, donde se tuvo en cuenta el error al valor final deseado. Además de esto, se crean 30 puntos de consulta como posiciones iniciales del movimiento, cuyos valores finales corresponden a los puntos de consulta creados para la comparación anterior. Los resultados obtenidos en el valor final no varían, y el desempeño de la técnica en cuanto al error de la posición inicial es mayor, si se compara con los errores que se obtienen con la técnica basada en distancia de Mahalanobis y distribución gaussiana.

La técnica DMP y GPR presentada tiene aplicaciones para el agarre de objetos en el espacio de trabajo, y reproducción de tareas que involucran cambios en las acciones de acuerdo a características del entorno, por ejemplo, al pintar un automóvil, donde la cantidad de pintura que se aplica varía según la zona que se desea pintar, o para codificar información de la fuerza de agarre con respecto al tipo de objeto o su orientación.

Se propone como trabajo futuro, la implementación de las técnicas DMP y GPR para generalización a nuevas trayectorias, sobre un brazo robótico real. Además, después de llevar la mano a la posición objetivo, capturar información del agarre, que pueda ser reproducida luego por el efector final del brazo robótico y que puede codificarse utilizando la técnica de aprendizaje por imitación presentada en este trabajo.

REFERENCIAS

- De Maesschalck, R., Jouan-Rimbaud, D., and Massart, D. L. (2000). The mahalanobis distance. *Chemometrics and intelligent laboratory systems*, 50 (1), pp. 1–18.
- Forte, D., Gams, A., Morimoto, J., and Ude, A. (2012). Online motion synthesis and adaptation using a trajectory database. *Robotics and Autonomous Systems*, 60 (10), pp. 1327–1339.
- Forte, D., Ude, A., and Gams, A. (2011). Real-time generalization and integration of different movement primitives. In *2011 11th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 590–595.
- Gams, A. and Ude, A. (2009). Generalization of example movements with dynamic systems. In *2009 9th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 28–33.
- Kaneko, T., Ono, T., and Munakata, N. (2011). Implementation of context-adaptive physical imitation between humans and robots. In *2011 IEEE RO-MAN*, pp. 187–191.
- Kober, J., Oztop, E., and Peters, J. (2011). Reinforcement learning to adjust robot movements to new situations. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, volume 22, pp. 2650.
- Kormushev, P., Calinon, S., and Caldwell, D. G. (2011). Imitation learning of positional and force skills demonstrated via kinesthetic teaching and haptic input. *Advanced Robotics*, 25 (5): 581–603.
- Kormushev, P., Calinon, S., Saegusa, R., and Metta, G. (2010). Learning the skill of archery by a humanoid robot icub. In *2010 10th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pages 417–423.
- Kulvicius, T., Ning, K., Tamosiunaite, M., and Worgotter, F. (2012). Joining movement sequences: Modified dynamic movement primitives for robotics applications exemplified on handwriting. *IEEE Transactions on Robotics*, 28 (1): 145–157.
- Lee, A. X., Huang, S. H., Hadfield-Menell, D., Tzeng, E., and Abbeel, P. (2014). Unifying scene registration and trajectory optimization for learning from demonstrations with application to manipulation of deformable objects. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014)*, pages 4402–4407.
- León, A., Morales, E., Altamirani, L., and Ruiz, J. (2011). Teaching a robot new tasks through imitation and on-line feedback. In *Proc. of the 16th Iberoamerican Congress on Pattern Recognition*.
- Lopes, M., Melo, F., Montesano, L., and Santos-Victor, J. (2010). Abstraction levels for robotic imitation: Overview and computational approaches. In Sigaud, O. and Peters, J., editors, *From Motor Learning to Interaction Learning in Robots*, volume 264 of *Studies in Computational Intelligence*, pages 313–355. Springer Berlin / Heidelberg.
- Lopes, M. and Santos-Victor, J. (2007). A developmental roadmap for learning by imitation in robots. *IEEE*

- Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 37 (2): 308–321.
- Metta, G., Sandini, G., Vernon, D., Natale, L., and Nori, F. (2008). The icub humanoid robot: an open platform for research in embodied cognition. In *Proceedings of the 8th Workshop on Performance Metrics for Intelligent Systems*, PerMIS '08, pp. 50–56, New York, NY, USA. ACM.
- Natale, L., Nori, F., Metta, G., Fumagalli, M., Ivaldi, S., Pattacini, U., Randazzo, M., Schmitz, A., and Sandini, G. (2013). The icub platform: a tool for studying intrinsically motivated learning. In *Intrinsically motivated learning in natural and artificial systems*, pp. 433–458. Springer.
- Nath, V. K. and Levinson, S. E. (2012). Learning to fire at targets by an icub humanoid robot. *Urbana*, 51, pp. 61801.
- Nemec, B. and Ude, A. (2012). Action sequencing using dynamic movement primitives. *Robotica*, 30(5), pp.837.
- Nguyen-Tuong, D., Seeger, M., and Peters, J. (2009). Model learning with local gaussian process regression. *Advanced Robotics*, 23(15), pp.2015–2034.
- Nicolescu, M. N. and Mataric, M. J. (2005). Task learning through imitation and human-robot interaction. In *Models and Mechanisms of Imitation and Social Learning in Robots, Humans and Animals: Behavioural, Social and Communicative Dimensions*, pp. 407–424. University Press.
- Oikonomidis, I., Kyriazis, N., and Argyros, A. (2011). Efficient model-based 3d tracking of hand articulations using kinect. In *BMVC 2011*. BMVA.
- Pastor, P., Hoffmann, H., Asfour, T., and Schaal, S. (2009). Learning and generalization of motor skills by learning from demonstration. In *2009 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 763-768.
- Peña-Solórzano, C.; Hoyos-Gutiérrez, J.; Prieto-Ortiz, F. (2015). Hacia el agarre de objetos utilizando aprendizaje robótico por imitación y datos de fuerza. *Revista EIA*, 12 (23) Enero-Junio, pp. 71-82.
- Rasmussen, C. E. (2004). Gaussian processes in machine learning. In *Advanced Lectures on Machine Learning*, pp. 63–71. Springer.
- Stulp, F., Theodorou, E., Buchli, J., and Schaal, S. (2011). Learning to grasp under uncertainty. In *2011 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5703–5708.
- Tamosiunaite, M., Nemec, B., Ude, A., and Wörgötter, F. (2011). Learning to pour with a robot arm combining goal and shape learning for dynamic movement primitives. *Robotics and Autonomous Systems*, 59 (11), pp. 910–922.
- Tan, H., Erdemir, E., Kawamura, K., and Du, Q. (2011). A potential field method-based extension of the dynamic movement primitive algorithm for imitation learning with obstacle avoidance. In *2011 International Conference on Mechatronics and Automation (ICMA)*, pp. 525–530.
- Tikhanoff, V., Cangelosi, A., and Metta, G. (2011). Integration of speech and action in humanoid robots: icub simulation experiments. *IEEE Transactions on Autonomous Mental Development*, 3 (1), pp. 17–29.
- Toda, Y., Kodai, Y., Hiwada, E., and Kubota, N. (2011). Human motion tracking for cognitive rehabilitation in informationally structured space based on sensor networks. In *2011 IEEE International Conference on Fuzzy Systems (FUZZ)*, pp. 1459 –1465.
- Ude, A., Gams, A., Asfour, T., and Morimoto, J. (2010). Task-specific generalization of discrete and periodic dynamic movement primitives. *IEEE Transactions on Robotics*, 26 (5), pp. 800–815.

**PARA CITAR ESTE ARTÍCULO /
TO REFERENCE THIS ARTICLE /
PARA CITAR ESTE ARTIGO /**

Peña Solórzano, C.A.; Hoyos Gutiérrez, J.G.; Prieto Ortiz, F.A.; (2018). Generalización de las trayectorias de un brazo robótico utilizando primitivas de movimiento dinámico y regresión de procesos gaussianos. *Revista EIA*, 15(29), enero-junio, pp. 109-123. [Online]. Disponible en: <https://doi.org/10.24050/reia.v15i29.690>