

# Programación por Demostración del Agarre de Objetos Empleando Técnicas Basadas en Modelo de Mezcla de Gaussianas

## Programming by Demonstration of Object Grasping Using Gaussian Mixture Model Based Techniques

José Gabriel : Hoyos Gutiérrez, Alvaro Andres Navarro Perez

Ingeniería eléctrica, Grupo de Investigación: GAMA, Universidad del Quindío, Armenia, Colombia

Correo-e: [aanavarro@uniquindio.edu.co](mailto:aanavarro@uniquindio.edu.co), [josegabrielh@uniquindio.edu.co](mailto:josegabrielh@uniquindio.edu.co)

**Resumen**—Se presenta una técnica que permite agarrar un objeto empleando la combinación de técnicas de programación por demostración. Las técnicas son el modelo de mezcla de gaussianas parametrizado en la tarea y el modelo de mezcla de gaussianas. Con la primera se estima la trayectoria de la mano para una nueva posición y orientación del objeto, el valor final obtenido de posición y orientación no siempre es exacto, por lo que se aplica un segundo paso o paso de corrección empleando el modelo de mezcla de gaussianas. A través de un experimento real se obtiene una base de datos de demostraciones de agarres y se prueba el algoritmo propuesto. Luego empleando evaluación subjetiva de la calidad del agarre y gráficas del error se muestra que la técnica presentada mejora la respuesta lograda al solo emplear el modelo de mezcla de gaussianas parametrizado en la tarea.

**Palabras clave**— Robótica, Programación por demostración de robots, agarre robótico, modelos de mezcla de Gaussianas.

**Abstract**— A programming by demonstration techniques combination that allows object grasping is presented. The techniques are the task-parameterized Gaussian mixture model and the Gaussian mixture models. With the first technique, a path hand to the new position and orientation of the object is estimated. Nevertheless, the final value obtained of position and orientation is not always accurate, so it encourages to apply a second method or step correction using Gaussian mixture models. A real experiment allow us to obtain a database of different grasps to subsequently validate the proposed algorithm. The obtained results by using a grasp quality subjective evaluation and graphics error show an improvement to the result generated only by using the task-parameterized Gaussian mixture model.

**Key Word** —Robotics, Robot programming by demonstration, robotic grasping.

### I. INTRODUCCIÓN

En programación por demostración (PpD) [1], [2], [3], [4], el robot es programado a partir de demostraciones de una tarea, con estas se obtienen modelos matemáticos, como por ejemplo el Modelo de Mezcla de Gaussianas (GMM sigla en inglés) [5]. Una mejora a la técnica anterior y que permite generalizar la reproducción de la tarea para una nueva posición del objeto, es el Modelo de Mezcla de Gaussianas Parametrizado en la Tarea (TP-GMM sigla en inglés) [6]. Con los modelos es posible recuperar u obtener incluso variaciones en la ejecución de la tarea, empleando la regresión de mezcla de Gaussianas (GMR sigla en inglés).

La primera tarea que debe realizarse para un buen agarre es posicionar y orientar la mano. La acción de realizar un buen agarre es un problema complejo y ha sido analizado desde los inicios de la robótica, siendo hoy en día un problema abierto. Aunque existen varios tipos de agarre, los dos principales son el agarre de potencia y el agarre de precisión [7]. En el primero el objeto es envuelto por la palma y los dedos, se utiliza para objetos pesados y/o de volumen mediano. En el segundo las puntas de los dedos realizan la sujeción y generalmente se utiliza para objetos livianos y/o de pequeño volumen.

El problema del agarre se puede dividir en dos: i) de correspondencia, y ii) de calidad del agarre. En el primer caso existen diferencias entre la mano humana y la mano del robot, por lo que se debe encontrar la correspondencia entre ambas. En el segundo caso, dado un objeto novedoso, se deben encontrar puntos de contacto de tal manera que el objeto pueda ser levantado y manipulado sin salirse de la mano. En la Figura 1, se muestra un ejemplo del proceso de agarre en PpD, a partir de obtener las trayectorias de articulación de los dedos y la posición y orientación (POSE) de la mano, se realiza un modelado de estas trayectorias, luego las salidas del modelo, deben ser mapeadas a las articulaciones disponibles

del robot (correspondencia), para finalmente, realizar la reproducción del agarre. Además de la codificación o modelado de las trayectorias de articulación del brazo y mano, también se ha considerado el problema de las fuerzas de contacto [8], las cuales evitan que el objeto se deslice. La calidad de un agarre depende de las fuerzas que aplica la mano robótica al objeto y de la forma en que lo rodea (depende de la posición y orientación de la mano), varios autores han planteado métricas para cuantificar la calidad como en [9], [10].

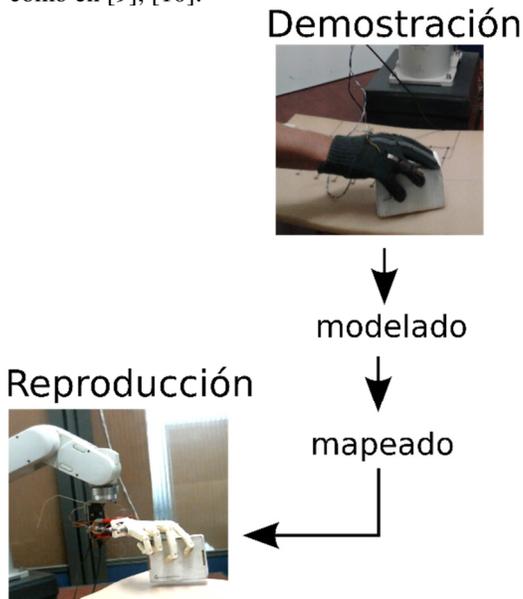


Figura 1. Diagrama de programación por demostración del proceso de agarre.

En 2009 Kragic y otros [12] presentaron una técnica de agarre de objetos por una mano robótica simulada. A partir de las demostraciones de las acciones de agarre, la clasificación de estos en diez tipos (por ejemplo: precisión, potencia) y el reconocimiento de los objetos a agarrar, generan una base de datos. Luego dado un objeto, una técnica de búsqueda simple que involucra métricas de calidad del agarre, recupera el tipo de agarre más apropiado. El anterior trabajo es continuado en 2010 [13], también usando una mano simulada, en este utiliza redes bayesianas, donde la estructura de la red, enlaza las variables: tipo de tarea, características del objeto, restricciones del agarre y acción de agarre. Una vez el proceso de entrenamiento de la red finalice, las dependencias entre las variables son aprendidas dando como resultado, que en el momento de presentarle un objeto específico a la red, esta genera la respectiva acción de agarre. Como desventaja se puede mencionar que la red bayesiana es implementada de forma manual requiriendo múltiples ejemplos para generar una respuesta aceptable.

En 2015 Song, Kragic y otros [14] se propuso una técnica donde mediante una red bayesiana se aprende la estructura de esta, a partir de la información de la manipulación de diferentes agarres de los objetos y sus restricciones. Con lo

anterior la red puede inferir el agarre y algunas tareas simples de manipulación. Los datos deben ser discretizados previo a la estimación de la red, para luego estimar las probabilidades de los enlaces entre las variables. El sistema es basado en la tarea, por lo tanto, las trayectorias de demostración no son aprendidas. Para tener una buena estimación de la red, se hace necesario una buena cantidad de agarres, así como un tiempo mediano de estimación.

En 2011 Balaguer y otros [15] propusieron una técnica donde utilizan visión estéreo, un brazo y garra Barret. El problema de agarre es dividido en dos etapas: Encontrar los puntos de contacto óptimo y Encontrar la POSE ideal de agarre. El sistema es capaz de agarrar objetos novedosos que guarden similitud con los demostrados. La desventaja radica en las demostraciones de agarre las cuales son tomadas directamente del robot y no a partir de imágenes de la mano humana.

Algunas técnicas combinan las trayectorias demostradas de posición de los dedos y las fuerzas que estos ejercen sobre el objeto, como por ejemplo [8], [16], [17]. En 2011 Schmidts y otros [8] capturan los datos de la tarea de agarre con: Un guante de datos, un sistema magnético de movimiento y un sistema háptico que obtiene la fuerza de cada dedo. En vez de usar las fuerzas directamente, se calculan las tensiones que interactúan entre los dedos y el objeto. Los autores reportan que la técnica mejora el agarre en casos donde el objeto cambia de tamaño o consistencia. Su desventaja está en que la técnica se aplica a un solo objeto (pelota). En 2012 Lin y otros [16], combinan control servo visual ojo en mano con las fuerzas demostradas para agarrar el objeto. La medición de la fuerza de agarre la hacen detectando el color de la uña de los dedos y codificándola con GMM/GMR. A diferencia de [8] sí usan las señales de fuerza directamente. Su desventaja está en que las cámaras usadas para mirar la uña, requieren ser ubicadas de manera específica.

Li y otros [17], propusieron en 2014, una técnica para agarrar y manipular objetos en tareas que requieren variar la fuerza y POSE (posición y orientación) de los dedos durante el transcurso de la tarea, como, por ejemplo, evitar el deslizamiento de una copa. A partir de demostraciones de agarres estimaron un modelo GMM que relaciona la fuerza y la POSE, para luego reproducir el movimiento.

En 2013 Huang y otros [18], presentaron una técnica de agarre que trabaja en tiempo real. A partir de múltiples agarres demostrados, se obtiene un modelo probabilístico GMM de estos. Con este modelo es posible calcular el agarre a partir de una posición y orientación inicial de la mano, con lo cual el modelo da como respuesta la posición, orientación y valores de articulación de los dedos finales. Los autores solo emplean algunos objetos en el entrenamiento y no plantean que ocurriría ante objetos novedosos, además se requiere una buena cantidad de agarres demostrados para su cálculo.

En este trabajo se emplean dos técnicas de programación por demostración llamadas TP-GMM y GMM. La primera permite generar la trayectoria de la mano para una nueva posición y orientación del objeto. Debido a que la POSE final no siempre es exacta, se recurre a la segunda técnica GMM donde se aplica un paso de corrección. Este modelo GMM se estima con tres datos:

- La POSE del objeto.
- La diferencia entre la POSE de la mano final de la trayectoria respuesta del TP-GMM y la POSE de la mano para cada uno de los puntos.
- Los valores de articulación final de los dedos para tomar el objeto.

En la reproducción del modelo GMM para una nueva posición y orientación del objeto, se entrega la corrección de la POSE y los valores de articulación a reproducir a través de una regresión (GMR). Aunque el trabajo presentado aquí guarda similitudes con [18], este genera toda la trayectoria del agarre y no solo la POSE final, además se necesitan menos puntos de demostración.

El resto del documento es como sigue, en la Sección II se presenta la técnica propuesta, en la Sección III se presentan los resultados. Por último, se presentan las conclusiones.

## II. TÉCNICA PROPUESTA

Se describen brevemente las técnicas GMM y TP-GMM, los parámetros de la tarea para estimar el modelo TP-GMM, el algoritmo implementado y el experimento con un robot real.

### A. Modelo de Mezcla de Gaussianas (GMM)

Es un modelo probabilístico que permite codificar múltiples puntos de datos  $\xi$  de dimensión  $D$  cada uno, a un modelo  $\Theta$  compuesto por  $K$  gaussianas, cada una de ellas con parámetros:  $\alpha_i$  coeficiente de mezcla,  $\mu_i$  vector de centros y  $C_i$  matriz de covarianza de la  $i$ -ésima gaussiana.

Los valores iniciales de los parámetros se obtienen, generalmente, usando el algoritmo de  $k$ -medias de los datos  $\xi$ , también se han usado valores iniciales obtenidos al realizar particiones iguales a la cantidad de gaussianas  $K$ , en el tiempo de los datos [5]. Luego estos valores son refinados usando el algoritmo EM (Esperanza-Maximización) [5]. La Regresión de Mezcla de Gaussianas (GMR) permite recuperar un dato completo a partir de información parcial de este, empleando los parámetros del modelo.

### B. Modelo de Mezcla de Gaussianas Parametrizado en la Tarea (TP-GMM)

Calinon y otros en 2012 [6], presentaron el modelo de mezcla de gaussianas parametrizado en la tarea. Este se puede ver como el resultado ponderado de varios modelos GMM y

ponderado según unos parámetros de la tarea, que son las posiciones y orientaciones de los objetos o puntos relacionados con la tarea. En la Fig. 2 se detallan los dos parámetros de la tarea empleados  $p_{11}$ :POSE inicial de la mano robot y  $p_{21}$  : POSE del objeto.

El modelo se estima a partir de  $M$  trayectorias demostradas, cada una con  $T_m$  puntos. Para cada punto es necesario conocer  $N_p$  parámetros de la tarea (marcos de referencia), que consisten en posiciones  $b_{n,j}$  y matrices de orientación  $A_{n,j}$ . Los índices  $n$  y  $j$  representan la muestra de tiempo  $n$  y el marco  $j$ -ésimo de referencia.

En la reproducción del modelo se emplea GMR. Los parámetros de la tarea para el proceso se enumeran a continuación: La POSE inicial del efector final, la POSE del objeto y el eje de tiempos de trayectorias. Una vez este proceso concluye, se conoce una nueva trayectoria cartesiana. En [19] puede encontrarse una descripción más detallada sobre el modelo TP-GMM.

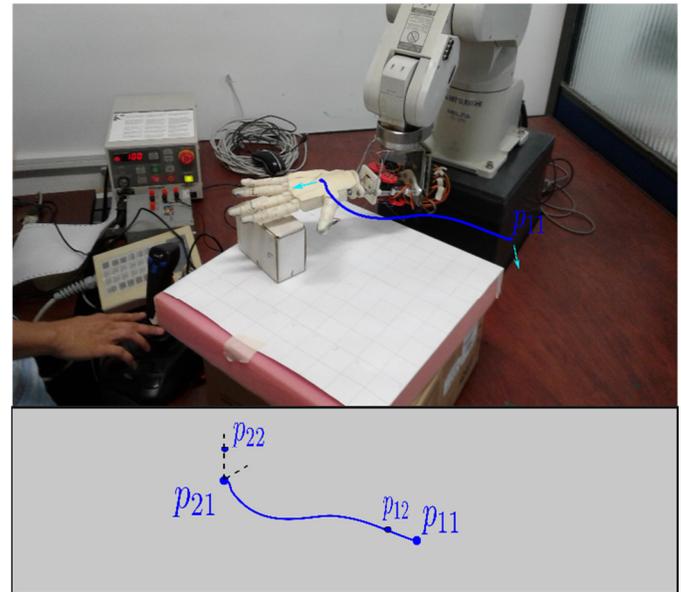


Figura 2. Parámetros de la tarea para el modelo TP-GMM.

### C. Obtención de los Parámetros de la Tarea

Para obtener los parámetros de la tarea  $A$  y  $b$ , se empleó un procedimiento similar al empleado en [19]. La Fig. 2 se muestran los puntos  $p_{21}$  y  $p_{22}$  para el parámetro de la tarea final. En el caso de la estimación de  $p_{12}$  se obtiene tres muestras desplazado del punto inicial  $p_{12}$ , y para la estimación de  $p_{22}$  se desplaza 3 cm en el eje  $z$  el punto final  $p_{21}$ , el valor de 3 cm fue seleccionado de manera heurística y hace que la trayectoria tenga un tramo final perpendicular al objeto.

Con estos puntos y el procedimiento comentado, se emplean (1) y (2):

$$\mathbf{b} = \begin{bmatrix} \mathbf{t} \\ \mathbf{p}_{a1,x} \\ \mathbf{p}_{a1,y} \\ \mathbf{p}_{a1,z} \\ \alpha \end{bmatrix} \quad (1)$$

con  $a$  uno o dos (Ej.  $a = 1$ ,  $\mathbf{p}_{11,x}$ ) y  $\alpha$  la orientación del robot o el objeto según el parámetro. Para el cálculo de  $\mathbf{A}$ , solo se usan los valores de posición ( $x, y, z$ ).

$$\mathbf{A} = \begin{bmatrix} \mathbf{1} & \mathbf{0}_{1 \times 3} & \mathbf{0}_{1 \times 4} \\ \mathbf{0}_{3 \times 1} & \mathbf{v} & \mathbf{0}_{3 \times 4} \\ \mathbf{0}_{4 \times 1} & \mathbf{0}_{4 \times 3} & \mathbf{I}_{4 \times 4} \end{bmatrix} \quad (2)$$

Donde  $\mathbf{v} = [\mathbf{v}_1 \ \mathbf{v}_2 \ \mathbf{v}_3]$ ,  $\mathbf{v}_2$  es el vector normal a la diferencia  $\mathbf{p}_{a2} - \mathbf{p}_{a1}$  y los vectores  $\mathbf{v}_1$  y  $\mathbf{v}_3$  son perpendiculares entre sí y con  $\mathbf{v}_2$ .  $\mathbf{A}$  representa la matriz de transformación, que contiene información de la orientación del punto  $\mathbf{p}_{a1}$  con respecto al punto  $\mathbf{p}_{a2}$ . Para el caso de reproducción de nuevas trayectorias, el punto  $\mathbf{p}_{a2}$ , se hallaría desplazado 3 cm en el eje  $z$  del punto final  $\mathbf{p}_{21}$ , el cual puede ser obtenido por visión de máquina.

#### D. Algoritmo

Se divide en la estimación de los modelos y la ejecución de estos:

##### Estimación:

Para la estimación del modelo TP-GMM, los parámetros de la tarea que se emplearon fueron dos: el primero es la posición y orientación inicial de la mano y el segundo es la posición y orientación del objeto.

En el caso de GMM, la estimación del modelo emplea tres datos:

- La POSE del objeto  $\mathbf{P}_{obj}$ .
- La diferencia  $\delta\mathbf{P}_r$  entre la POSE final respuesta dada por el TP-GMM y la POSE de la mano para cada uno de los puntos.
- Los valores de articulación de los dedos  $\mathbf{q}$ .

Los valores de posición y orientación de la mano, se lograron al variar la posición y orientación del objeto para  $n$  puntos. Para cada punto se ejecuta el modelo TP-GMM y se calcula la diferencia. Luego con GMR y el modelo GMM es posible recuperar el valor de diferencia y la articulación (7):

$$[\delta\mathbf{P}_r, \mathbf{q}] = GMR(\Theta, \mathbf{P}_{obj}). \quad (7)$$

##### Ejecución:

El algoritmo de ejecución, se compone de los siguientes pasos:

#### 1) Ubicación inicial.

- Obtención de la posición y orientación del objeto.
- Cálculo de los parámetros de la tarea.
- Cálculo de la trayectoria usando GMR y el modelo TP-GMM.
- Ejecución de la trayectoria y obtención de la

POSE de la mano final previa.

#### 2) Corrección.

- A partir del modelo GMM se estima la diferencia de POSE  $\delta\mathbf{P}_r$  según la posición y orientación del objeto.
- Cálculo y ejecución del nuevo valor de POSE  $\mathbf{P}_r = \mathbf{P}_r + \delta\mathbf{P}_r$ .

La mano realiza el agarre, según los valores de articulación  $\mathbf{q}$  entregados por el modelo GMM empleando GMR.

#### E. Experimento

Se realizó un experimento, que consistió en el empleo de un robot industrial Mitsubishi RV2AJ, una mano de desarrollo artesanal y una palanca de juego (joystick), con estos se obtuvieron varios puntos de agarre y trayectorias demostradas. Las comparaciones se realizaron empleando el error y una evaluación subjetiva de la calidad del agarre.

El RV2AJ es un robot de cinco grados de libertad, al cual se le agregó una mano de 5 grados de libertad basada en el desarrollo presentado en [20] (Figura 4) impresa en plástico ABS en tres dimensiones. El movimiento del robot en la fase de demostración se hace empleando una palanca de juego. Con la cual se mueve de manera incremental en los ejes ( $x, y$ ) y con la rueda en el eje  $z$ , la combinación de la pulsación de un pulsador de la palanca y la rueda se emplea para la orientación de la mano. El movimiento de los dedos realizado en cada demostración, fue obtenido a partir de sensores de flexión, instalados sobre un guante de datos de fabricación artesanal [21]. Para cada posición de agarre se empleó una caja pequeña como se muestra en la Figura 1. Moviendo el robot con la palanca de juego se obtuvieron 5 trayectorias de demostración para 5 ubicaciones de la caja y por otro lado 84 puntos de agarre, pero sin trayectoria.

De cada uno de estos se obtuvo un dato compuesto por: La POSE del objeto, la POSE del robot. Los 84 puntos de agarre corresponden a una cuadrícula de  $8 \times 4$  y 3 orientaciones (0, 30, -45 grados) por cada uno, de los cuales, debido a los límites del espacio de trabajo del robot, no se alcanzaron 12 puntos. La selección de las cinco trayectorias se realizó de manera heurística buscando que cubriera las diversas posiciones y orientaciones. Con estas se estima el modelo TP-GMM y con los 84 puntos el modelo GMM. Con lo anterior, se ejecutó el algoritmo presentado anteriormente y se obtuvieron los resultados.

### III. RESULTADOS

Dado que es complejo calcular una métrica de calidad como en [22], debido a que no se cuenta con las fuerzas de las caras de las falanges por cada dedo, se calculó el error como la diferencia entre la POSE demostrada y la estimada con TP\_GMM o la técnica propuesta. En la Figura 5 se muestra el

error tanto para la técnica TP-GMM en color claro y asteriscos, como para la técnica propuesta en color oscuro y círculos.

(derecha). El número indica el punto demostrado de agarre de los 84 realizados.



Figura 4. Robot RV2AJ y mano.

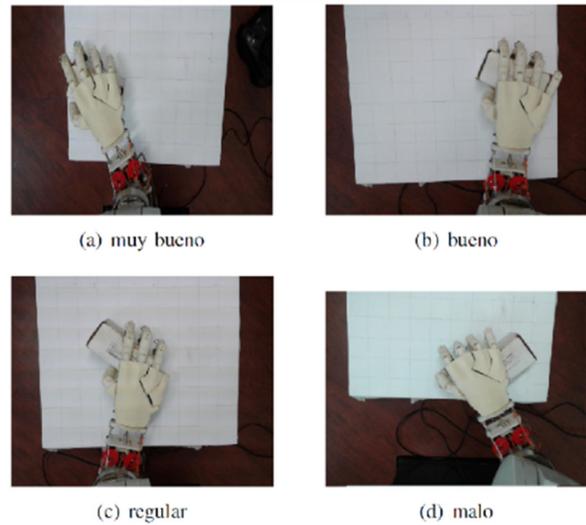


Figura 6. Ejemplos de agarres finales a) muy bueno, b) bueno, c) regular d) malo.

En la Figura 6 se muestran ejemplos de agarres que se clasificaron de manera subjetiva como: a) muy bueno, b) bueno, c) regular, d) malo. Las anteriores clases se asignaron según el grado de cubrimiento del objeto por la mano y la estabilidad que pueda tener el agarre.

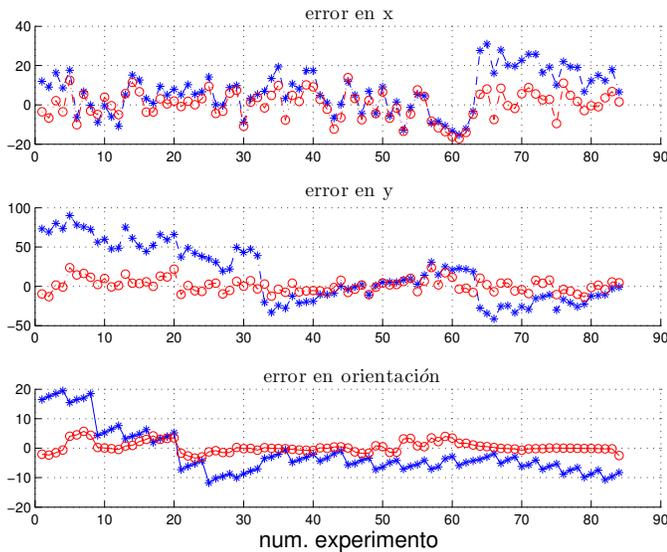


Figura 5. Gráfica del error usando TP-GMM (asteriscos) y la técnica propuesta (círculos), para las pruebas con el robot RV2AJ.

A partir de lo anterior y dando como segundo parámetro de la tarea la posición y orientación de la caja empleada en las pruebas, se obtuvo la gráfica de la Figura 7 para 36 puntos. En línea clara se muestra el resultado al solo emplear TP-GMM y en línea oscura la técnica propuesta.

En la Figura 8, se presentan algunas imágenes que permiten contrastar la respuesta de POSE final, usando solo TP-GMM (izquierda) y la obtenida empleando la técnica propuesta

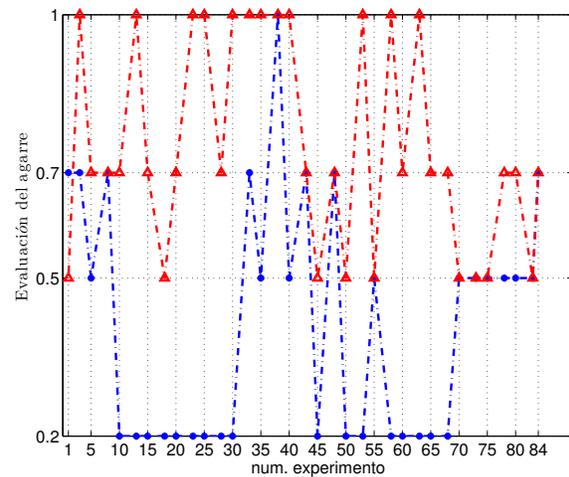


Figura 7. Gráfica de la evaluación subjetiva para 36 puntos de agarre de los 84 puntos empleados en la estimación. En línea roja el resultado de solo TP-GMM y en línea azul la técnica propuesta. (1.0 muy bueno, 0.7 bueno, 0.5 regular, 0.2 malo.)

De la Figura 5 es posible observar que el error se reduce a valores bajos en el eje Y y en la orientación, por otro lado, de la evaluación subjetiva mostrada en la Figura 7, la calidad se mejora y en ninguno de los casos probados es calificada como mala para la técnica propuesta.

Si bien los resultados presentados son para el agarre de una caja, el algoritmo es simple de adaptar a otros objetos. En la Figura 9, se muestra el agarre de una bola y un tarro cilíndrico.

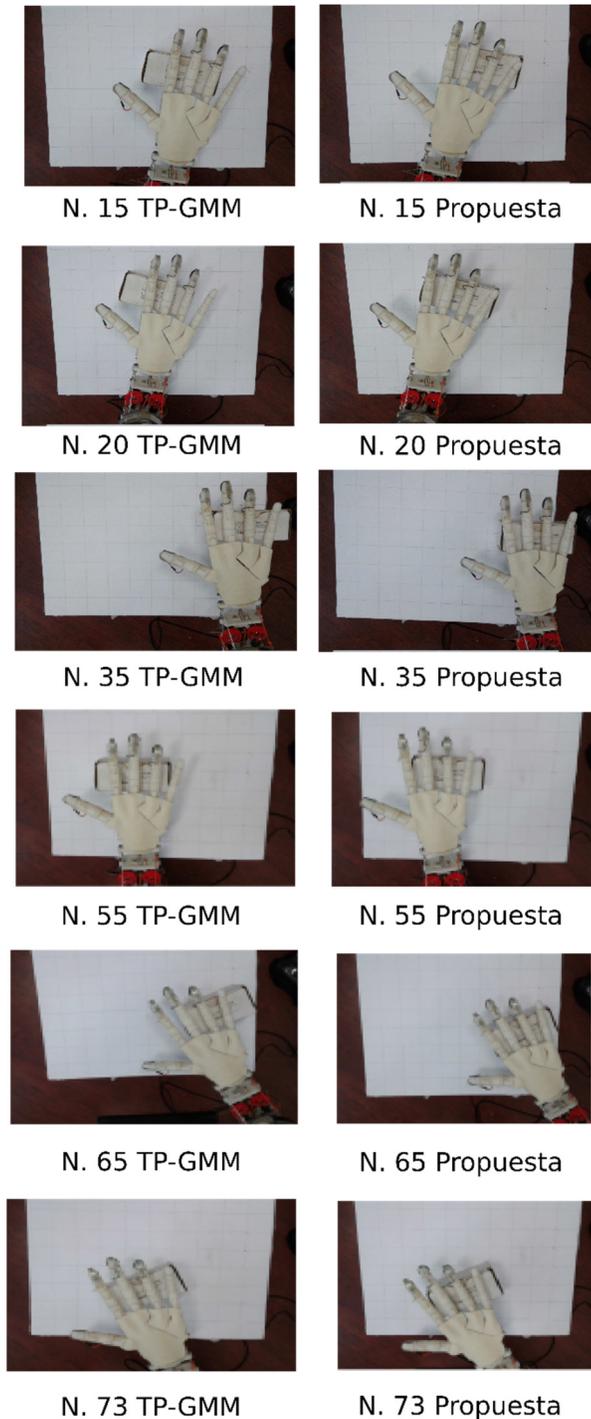


Figura 8. Imágenes para la respuesta de POSE final, usando solo TP-GMM (izquierda) y la propuesta (derecha). El número indica el punto demostrado de agarre de 84 realizados.



Figura 9. Ejemplo del agarre de otros objetos.

## CONCLUSIONES

Para un agarre confiable, se requiere posicionar y orientar la mano con respecto a la posición y orientación del objeto. En este trabajo se emplearon dos técnicas de modelos probabilísticos para: a) Generar una trayectoria desde un punto inicial hasta el objeto muy similar a las realizadas por un humano (TP-GMM), b) Mejorar la posición y orientación resultado de la técnica y dar los valores de articulación a ejecutar por los dedos.

La técnica TP-GMM requiere que los parámetros de la tarea estén relacionados con las trayectorias que modela, en algunos casos el punto que permite calcular la orientación de llegada de la trayectoria es de difícil obtención, por lo que la técnica propuesta ayuda a corregir el resultado de TP-GMM, como se puede observar de las Figuras 5 a 8.

Una de las limitantes para el agarre de los objetos es la cantidad de grados de libertad de la mano y que el pulgar no tenga los movimientos (flexión - extensión y abducción-aducción), por eso se planea desarrollar una mano más completa.

Como trabajo futuro, también se piensa emplear en la recolección de residuos que viajan por una banda transportadora.

## REFERENCIAS

- [1] S. Schaal, «Is imitation learning the route to humanoid robots?,» *Trends in cognitive sciences*, vol. 3, n° 6, pp. 233-242, 1999.
- [2] A. Billard, S. Calinon, R. Dillmann y S. Schaal, «Robot Programming by Demonstration,» de *Handbook of Robotics*, Springer, 2008.

- [3] B. D. Argall, S. Chernova, M. Veloso y B. Browning, «A survey of robot learning from demonstration,» *Robotics and Autonomous Systems*, vol. 57, n° 5, pp. 469-483, 2009.
- [4] C. Breazeal y B. Scassellati, «Robots that imitate humans,» de *trends in Cognitive Sciences*, 2002.
- [5] S. Calinon y A. Billard, «Incremental Learning of Gestures by Imitation in a Humanoid Robot,» de *IEEE Human Robot Interaction Conference*, 2007.
- [6] S. Calinon, Z. L. a. T. A. a. N. G. Tsagarakis y D. Caldwell, «Statistical dynamical systems for skills acquisition in humanoids,» de *8th IEEE-RAS International Conference on Humanoid Robots*, 2012.
- [7] M. Cutkosky, «On grasp choice, grasp models, and the design of hands for manufacturing tasks,» *Robotics and Automation, IEEE Transactions on*, vol. 5, n° 3, pp. 269-279, Jun 1989.
- [8] A. M. Schmidts, D. Lee y A. Peer, «Imitation learning of human grasping skills from motion and force data,» de *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, 2011.
- [9] A. T. Miller y P. K. Allen, «Graspit! A versatile simulator for robotic grasping,» *IEEE Robotics Automation Magazine*, vol. 11, n° 4, pp. 110-122, Dec 2004.
- [10] M. R. J. C. Raúl Suárez, «Grasp quality measures,» 2006.
- [11] K. Hsiao y T. Lozano-Perez, «Imitation Learning of Whole-Body Grasps,» de *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, 2006.
- [12] J. Tegin, S. Ekvall, D. Kragic, J. Wikander y B. Iliev, «Demonstration-based learning and control for automatic grasping,» *Intelligent Service Robotics*, vol. 2, n° 1, pp. 23-30, 2009.
- [13] D. Kragic, D. Song, K. Huebner y V. Kyrki, «Learning Task Constraints for Robot Grasping using Graphical Models,» de *IROS*, 2010.
- [14] D. Song, C. H. Ek, K. Huebner y D. Kragic, «Task-Based Robot Grasp Planning Using Probabilistic Inference,» *IEEE Transactions on Robotics*, vol. 31, n° 3, pp. 546-561, June 2015.
- [15] S. C. Benjamin Balaguer, «Human-Inspired Grasping of Novel Objects through Imitation Learning,» de *Workshop on Autonomous Grasping at IEEE ICRA*, 2011.
- [16] Y. Lin, S. Ren, M. Clevenger y Y. Sun, «Learning grasping force from demonstration,» de *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, 2012.
- [17] M. Li, H. Yin, K. Tahara y A. Billard, «Learning object-level impedance control for robust grasping and dexterous manipulation,» de *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014.
- [18] B. Huang, S. El-Khoury, M. Li, J. Bryson y A. Billard, «Learning a real time grasping strategy,» de *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, 2013.
- [19] J. Hoyos-Gutierrez, C. Peña-Solorzano, C. Garzon-Castro y F. Prieto-Ortiz, «Hacia el Manejo de una Herramienta por un Robot NAO Usando Programación por Demostración,» *revista tecnológicas, ITM, Colombia*, vol. 17, n° 33, pp. 65-76, 2014.
- [20] Langevin, «inmoov hand,» abril 2017. [En línea]. Available: <http://inmoov.fr/>.
- [21] C. Peña Solorzano, J. Hoyos Gutierrez y F. Prieto Ortiz, «Hacia El Agarre De Objetos Utilizando Aprendizaje Robótico Por Imitación Y Datos De Fuerza,» *Revista EIA*, vol. 12, n° Ed. N.23, pp. 71-82, Enero-Junio 2015.
- [22] J. Hoyos y A. Navarro, «Combination of two programming by demonstration techniques to improve the quality of a grasp,» de *2016 IEEE Colombian Conference on Robotics and Automation (CCRA)*, 2016.