

CONTROL DEL BRAZO ROBÓTICO DE SEIS GRADOS DE LIBERTAD, MEDIANTE RETROALIMENTACIÓN VISUAL, PARA LA COSECHA DEL AJÍ PAPRIKA

ROBOTIC ARM CONTROL SIX DEGREES OF FREEDOM THROUGH VISUAL FEEDBACK TO HARVEST OF PAPRIKA PEPPER

Ricardo Rodriguez B¹, Mark Sanchez P², Miguel Ravichagua I³, Sebastián Castillo H⁴.

RESUMEN

El presente trabajo es la culminación de la interacción mecatrónica de las áreas de visión artificial y control de robots, para la automatización del proceso de la cosecha del ají paprika. El proceso se lleva a cabo mediante la utilización de un robot de 6 grados de libertad, el cual brinda mayor versatilidad para el desarrollo de movimientos complejos y operaciones en terrenos agrestes. Para gobernar dichos movimientos y la posición espacial del robot se hace uso de un sensor visual, entregando las imágenes necesarias para reconocer el ambiente y un sensor de distancia para obtener la profundidad y así conseguir la ubicación específica del ají en su espacio de trabajo.

Palabras clave: Torque computado, Control por retroalimentación visual, Sensor visual.

ABSTRACT

The present document is the culmination of mechatronic interaction, between the computer vision and control of robot areas for automation process of paprika pepper's harvest. A robot of 6 DOF (degree of liberty) perform the process of harvest, this robot give to us more versatility in complex movements and operations on rural lands. To operate the movements and 3D positions of the robot we use a visual sensor and distance infrared sensor. The visual sensor gives us required images to recognize the environment, the sensor infrared measures the deep, and with these two values we can obtain the specific location of the peppers in the robot's workspace.

Keyword: Computed torque, Visual servoing, Visual sensor.

INTRODUCCIÓN

El advenimiento de la tecnología y su incesante desarrollo en aplicaciones de la ingeniería de control de los sistemas mecatrónicos, traen consigo innovadas técnicas de control lineal y no lineal para procesos multivariantes de varias entradas y varias salidas, alcanzando en los últimos años un elevado nivel de aceptación industrial en control de procesos. En nuestro caso, el sistema

multivariable, brazo robótico cosechador, se presenta como una alternativa de solución al método manual de cosecha. El brazo robótico cosechador es capaz de diferenciar automáticamente los frutos maduros y cosecharlos, con un daño mínimo al arbusto del ají paprika. Para el diseño del brazo robótico se optó por un modelo de 6 grados de libertad, para darle mayor libertad de movimiento en su espacio de trabajo.

¹Docente investigador de la Facultad de Ingeniería Mecánica de la Universidad Nacional de Ingeniería, ²Egresado de la Facultad de Ingeniería Mecánica de la Universidad Nacional de Ingeniería, ³Egresado de la Facultad de Ingeniería Mecánica de la Universidad Nacional de Ingeniería, ⁴Egresado de la Facultad de Ingeniería Mecánica de la Universidad Nacional de Ingeniería.

Además está equipado con un sensor visual, que permite registrar imágenes del entorno y; un sensor de distancia para medir la distancia del fruto respecto al efecto final del brazo robótico cosechador.

PROCESAMIENTO DE IMÁGENES

Para el reconocimiento de los frutos maduros, haremos uso de las técnicas de procesamiento de imágenes como el filtro HSV (Hue, Saturation Value) y detectar regiones conectadas en imágenes digitales binarias (Detección de Burbujas). Todo el cálculo se llevara a cabo en una computadora, que

a la vez será el comando central que gobierna los movimientos del brazo robótico.

En la Fig. 1 se puede observar que el primer proceso, es el filtro por color, en este caso usamos el filtro por HSV que consiste en cambiar de espacio de color, las fotos recibidas en RGB por el sensor visual. Este cambio de color se opta porque queremos obtener una tonalidad de rojo característica del fruto maduro, para nuestro caso rojo vino tinto. El filtro HSV es apropiado para nuestra tarea ya que permite mantener las características de color deseado a pesar de los cambios bruscos de luminosidad y brillo.

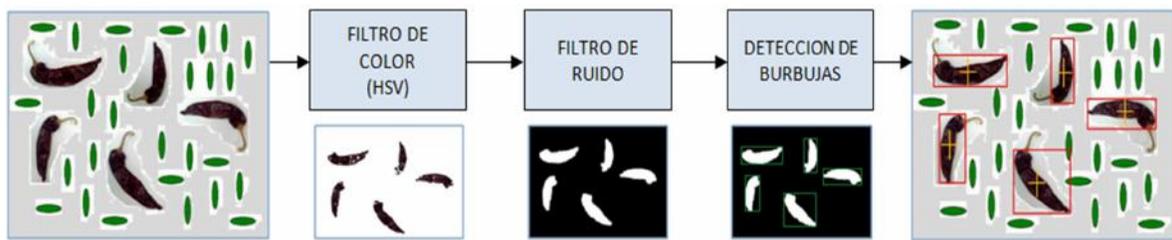


Fig. 1 Procesos principales para reconocimiento de ajes pprika maduros.

La siguiente etapa del reconocimiento de ajes es el filtro de ruido, por el cual quitamos pequeas imperfecciones producto del filtrado inicial, consideradas ruido de la imagen, esto lo hacemos mediante diversos tipos de filtro adaptativos.

La ultima etapa del reconocimiento es la deteccin de burbuja. Este algoritmo consiste en el agrupamiento de regiones que estn conformadas por grupos de píxeles cercanos entre s, dentro de una imagen binaria. Adems obtenemos con esta tcnica la orientacin del objeto con respecto a la vertical y el centroide con respecto a la esquina superior izquierda de la imagen.

CONTROL DE BRAZO ROBTICO

El algoritmo de control se asegura que el manipulador robtico pueda ejecutar sus movimientos de forma rpida y precisa. Se empezar mostrando el modelamiento matemtico y movimiento, expresando sus parmetros de Denavit-Hartenberg, a continuacin se presenta el esquema de control adoptado, se detallarn sus componentes y se presentarn los resultados de las simulaciones realizadas, las cuales, muestran al controlador en funcionamiento.

En la Fig. 2 se muestra un esquema del brazo robtico y los sistemas de coordenadas de cada articulacin

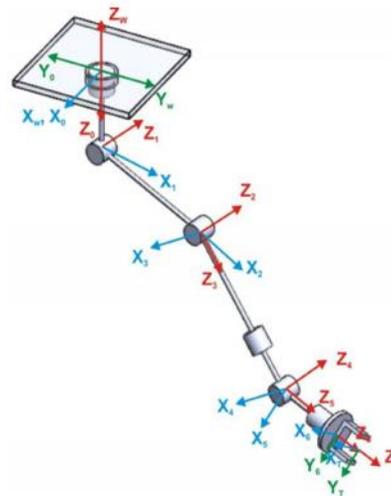


Fig. 2 Sistemas de coordenadas de Denavit-Hartenberg.

La tabla 1 muestra los parmetros de Denavit-Hartenberg para cada articulacin, los cuales son

necesarios para el análisis de la cinemática y dinámica del robot.

Tabla 1. Parámetros de Denavit-Hartenberg para el Manipulador.

Nº	d	θ	a	α
w-0	0	0	0	$\frac{\pi}{2}$
0-1	a_1	θ_1	0	$\frac{\pi}{2}$
1-2	0	θ_2	a_2	0
2-3	0	$\frac{\pi}{2} + \theta_3$	0	$\frac{\pi}{2}$
3-4	a_3	θ_4	0	$-\frac{\pi}{2}$
4-5	0	θ_5	0	$\frac{\pi}{2}$
5-6	a_4	θ_6	0	0

Torque computado

A continuación se explica la estrategia de control por torque computado.

Sea el modelo del robot rígido:

$$\tau = M(q)\ddot{q} + V(q, \dot{q}) + G(q) \quad (1)$$

Donde q es un vector de 6 dimensiones que representa al conjunto de las variables articulares, y \dot{q} y \ddot{q} su primera y segunda derivada con respecto al tiempo, $M(q)$ se denomina Matriz de Masas que depende sólo del valor de q , $V(q, \dot{q})$ es la Matriz de Fuerzas Centrifugas y de Coriolis, que depende de q y de \dot{q} , $G(q)$ es el Vector de Fuerzas Gravitacionales, y τ es el vector cuyas componentes representan los torques ejercidos por los motores. Asumiendo que se tiene acceso a las variables q y \dot{q} , el control por torque computado define la ley de control de la siguiente forma:

$$\tau = \hat{M}(q)(\ddot{q}_d - u) + \hat{V}(q, \dot{q}) + \hat{G}(q) \quad (2)$$

Donde $\hat{M}(q)$, $\hat{V}(q, \dot{q})$ y $\hat{G}(q)$ son aproximaciones a las matrices $M(q)$, $V(q, \dot{q})$ y $G(q)$, \ddot{q}_d es la segunda derivada de la trayectoria deseada, y u es la salida de algún otro controlador en lazo cerrado.

Reemplazando (2) en (1):

$$M(q)\ddot{q} + V(q, \dot{q}) + G(q) = \hat{M}(q)(\ddot{q}_d - u) + \hat{V}(q, \dot{q}) + \hat{G}(q) \quad (3)$$

$$\hat{M}(q)(\ddot{q}_d - u) + \hat{V}(q, \dot{q}) + \hat{G}(q)$$

Si asumimos que se conocen exactamente las matrices del modelo, es decir:

$$\begin{aligned} \hat{M}(q) &= M(q) \\ \hat{V}(q, \dot{q}) &= V(q, \dot{q}) \\ \hat{G}(q) &= G(q) \end{aligned} \quad (4)$$

Entonces la ecuación (3) se reduce a:

$$\begin{aligned} \ddot{q} &= \ddot{q}_d - u \\ u &= \ddot{q}_d - \ddot{q} \\ u &= \ddot{e} \end{aligned} \quad (5)$$

Donde $e = q_d - q$ es el error en el seguimiento de la trayectoria. De esta forma podemos definir la señal u como la salida de un controlador PD:

$$u = -K_v \dot{e} - K_p e \quad (6)$$

Reemplazando (6) en (5):

$$\ddot{e} + K_v \dot{e} + K_p e = 0 \quad (7)$$

La ecuación (7) representa 6 sistemas lineales estándar de 2do orden. Si las matrices K_v y K_p se definen como matrices diagonales, cada uno de estos 6 sistemas se desacopla y se puede tener 6 problemas de control clásicos:

$$\ddot{e} + K_v \dot{e} + K_p e = 0 \quad (8)$$

Usando la transformada de Laplace, se obtiene:

$$G(s) = s^2 + K_v s + K_p \quad (9)$$

Comparándolo con la forma estándar de un sistema de segundo orden:

$$G(s) = s^2 + 2\xi\omega_n s + \omega_n^2 \quad (10)$$

Se obtiene:

$$\begin{aligned} K_p &= \omega_n^2 \\ K_v &= 2\xi\omega_n \end{aligned} \quad (11)$$

Por lo general no se desea sobre impulsos debido a que puede hacer que el efector se salga de su trayectoria, por lo cual se tiene desde el principio que $\xi = 1$, y entonces:

$$\begin{aligned} K_p &= \omega_n^2 \\ K_v &= 2\omega_n \end{aligned} \quad (12)$$

El problema de control se reduce a escoger la frecuencia natural. Para este caso, $\omega_n = 10$ rad/s mostró ser adecuado. En el caso real, habrá una discrepancia entre el modelo real y las matrices que calculamos del robot, es decir, la ecuación (4) no cumplirá exactamente, y el control PD podría no ser suficiente para compensar esto. Es por eso que se agrega una pequeña componente integral al controlador, de forma que la ecuación (6) se modifica a:

$$u = -K_v \dot{e} - K_p e - K_i \int_0^t e dt \quad (13)$$

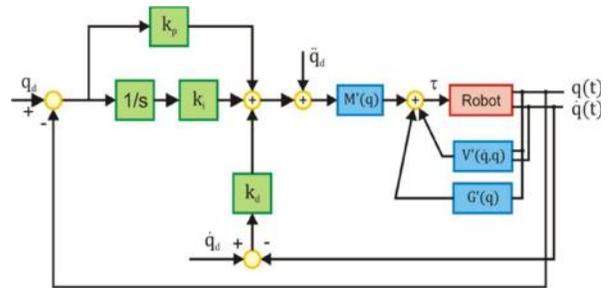


Fig. 3 Esquema del control por torque computado.

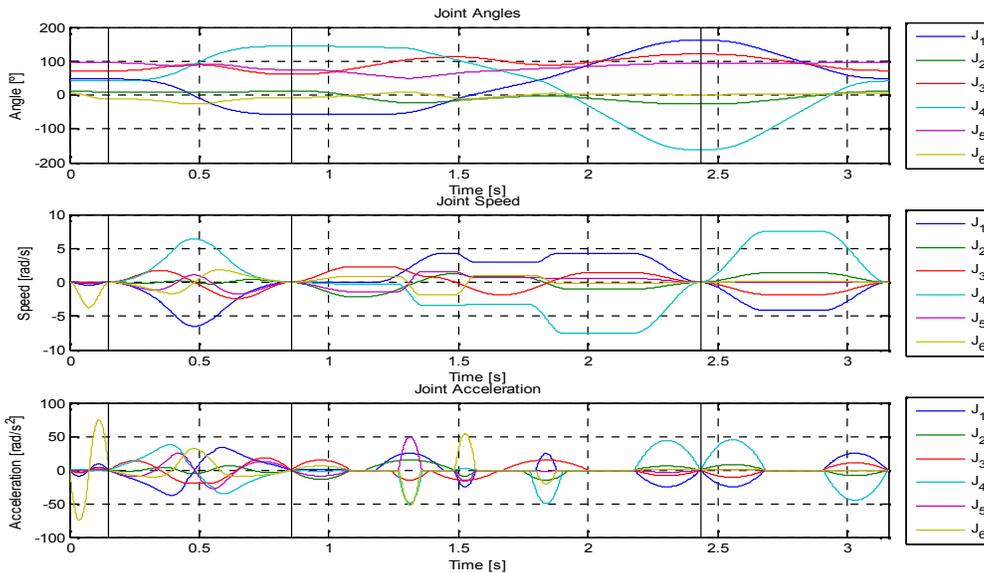


Fig.4 Evolución de los ángulos de las juntas para una de las trayectorias regulares.

La Fig. 3 muestra gráficamente las ecuaciones que acabamos de mostrar. Las componentes azules son las aproximaciones a las matrices $M(q)$, $V(q, \dot{q})$, $G(q)$, y las componentes verdes representan al controlador PID.

Podemos observar que las componentes azules solo miden el estado actual, pero no obtienen información del estado deseado. Por este motivo, se conocen como componentes enfeedforward.

En cambio, el controlador PID tiene su entrada a partir del error entre la trayectoria deseada y la que realmente se está obteniendo. Por este motivo a esta componente del controlador se le conoce como componente en feedback.

RESULTADOS

Las simulaciones del desempeño del controlador se realizaron usando la formulación de dinámica directa del robot basada en dinámica multicuerpo. La ecuación general del sistema bajo esta formulación está dada por:

$$M\ddot{q} + C_q^T \lambda = Q_e + Q_v \quad (14)$$

Donde M es la matriz de masas del sistema, Q_v es el vector de fuerzas de coriolis, Q_e es el vector de fuerzas externas incluida la gravedad, y $C_q^T \lambda$ es el vector de fuerzas debido a las restricciones (juntas). Nótese que estas matrices de masas y de

coriolis son diferentes a las matrices $M(q)$ y $V(q, \dot{q})$ definidas en la ley de control.

Se simuló usando varias trayectorias definidas para dimensionar los motores, y se encontró que en general el algoritmo de control funciona adecuadamente. La Fig. 4 muestra la evolución de los ángulos de las juntas para una de las trayectorias estándar.

CONTROL POR RETROALIMENTACION VISUAL

El control por retroalimentación permite manipular los movimientos del brazo robótico, mediante el uso de un sistema de visión – conformados de un sensor visual y un sensor de distancia – cosechar los ajíes paprika. La captura del video que genera el programa para visualizar el movimiento resultante podemos observar en la Figura 5.

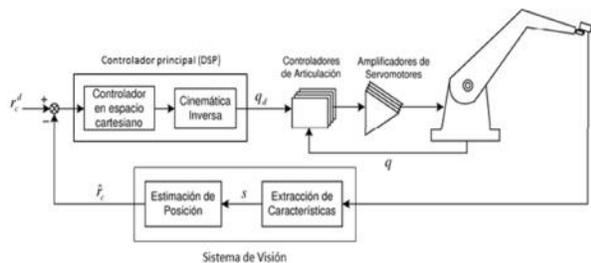


Fig. 5 Mira y Mueve – estático, basado en posición.

Existen diversas configuraciones de control por retro-alimentación visual, en nuestro trabajo se eligió una variante del “Mira y Mueve – Estático, basado en posición”.

La variante del “Mira y Mueve – Estático, basado en posición”, consiste en que el sistema de visión, calcula la posición del ají paprika usando dos movimientos de brazo robótico.

En el esquema de la Fig. 5, asumimos la entrada de referencia cero, puesto que el sistema de visión, entrega al controlador central (DSP) la posición aproximada del ají paprika.

Este controlador principal procesa el dato recibido y envía información a cada controlador de cada articulación, para que el efecto final del brazo robótico se mueva hacia la posición del ají paprika y pueda cosecharlo.

La variante de este esquema, está en el sistema de visión, puesto que usa dos movimientos para calcular la posición del ají. El primer movimiento es calcular la orientación del ají y el segundo es calcular la distancia del mismo.

Pruebas de laboratorio



Fig. 6 Brazo Robótico Iniciando cosecha de ají Páprika.



Fig. 7 Ají paprika cosechado por el brazo robótico y siendo llevado a una cesta.

CONCLUSIONES

Usando el filtro HSV se logra extraer el ají paprika de su entorno, pudiendo obtener con el algoritmo de detección de burbujas la orientación del fruto en el plano de la imagen.

Analizando los requerimientos para el control del manipulador, se determinó que la estrategia de torque computado es la más adecuada para esta aplicación.

Mediante la variante usada para el control vía retroalimentación visual usamos solo 2 movimientos, evitando de esta manera usar un sensado continuo y movimientos adicionales del robot, que consumirían más tiempo y por ende, se garantiza su operación.

Se logró coger una paprika en 3 segundos, es decir, sacarlo de su arbusto; esto sin considerar el tiempo de llevado de la paprika hacia su cesta donde se almacena.

La efectividad máxima de cosecha para una paprika es de dos a tres rutinas de cosecha del brazo robótico, esto debido a algunos problemas de precisión en el tema mecánico y electrónico.

El tiempo de procesamiento de imágenes para el reconocimiento de un ají paprika fue como

AGRADECIMIENTOS

máximo de 150 mil segundos.

A la empresa DIROSE S.A.C y al FINCyT, por haber financiado este proyecto.

REFERENCIAS

1. **Gonzales, R. C., Woods, R. E.**, "Tratamiento Digital de Imágenes". Addison-Wesley Iberoamericana, S.A., 1996.
2. <http://code.google.com/p/cvblob/>
3. **Luh, J. Y. S., Walker, M. W., Paul**, "On-Line Computational Scheme for Mechanical Manipulators", 1980, Journal of Dynamic Systems, Measurement, and Control, Vol. 102.
4. **Lewis, Frank L., Dawson, Darren M., Abdallah, Chaouki T.**, "Robot Manipulator Control - Theory and Practice", 2nd edition, 2004.

Correspondencia: robust@uni.edu.pe