

Desarrollo de un modelo para el seguimiento de patrones mediante reconocimiento óptico

Development of a model for pattern tracking by optical recognition.

Rodrigo Andrés Franco Luna¹, José Andrés Chaves Osorio²
 Departamento de Física, Universidad Tecnológica, Pereira, Colombia
 rodrigo_franco@utp.edu.co, jachaves@utp.edu.co

Resumen— En este documento se relata la experiencia del diseño de un sistema de identificación de patrones por medio del procesamiento de imágenes. En la parte inicial se detalla completamente el proceso de identificación de las características de interés (para este caso color) de un objeto que en adelante de le denominará objeto de prueba, posteriormente el algoritmo implementado entregará los datos de posición relativa de la imagen con respecto al centro de la misma, estos datos serán realimentados a un sistema de control de posición.

Palabras clave— Algoritmos de identificación, Control de posición, Parámetros, Posición relativa, Procesamiento de imágenes.

Abstract— This paper describes the experience of designing an identification system parameters via image processing, in the first part detailed the entire process of identification of parameters (for this case Color) of an object in front of him called the test object, then the algorithm implemented data give a position on the image to the center of it, these data will be fed back to a position control system.

Keywords — Identification algorithms, Control position, Parameters, Relative Position, Image Processing.

I. INTRODUCCIÓN

Los sistemas de seguimiento usando imágenes tienen una amplia aplicación en el mundo moderno, se usan en sistemas de seguridad, validación de acciones en deportes, televisión, etc. En la mayoría de los casos estos sistemas son operados directamente por seres humanos, lo cual implica que los resultados están sujetos a la subjetividad de la percepción humana, lo cual puede llegar a convertirse en un inconveniente en temas relacionados con la toma de decisiones (como ocurre en el caso de decisiones arbitrales), de allí que implementar dispositivos de decisión “más objetivos” traen grandes beneficios, como pueden ser,

detectar una presencia fuera de lo común en un establecimiento comercial lo que permite capturar un ladrón y evitar un robo, definir si un balón de fútbol traspasó en más del 50% la línea de meta y así aprobar un gol, o simplemente no perderse los mejores detalles de una presentación. En síntesis la situación que quiere resaltarse es que resulta más objetivo cuando un sistema electro-mecánico bajo unos parámetros predeterminados se encarga de tomar las decisiones (por ejemplo a través de la toma de imágenes).

El planteamiento del trabajo se basa en el desarrollo de un sistema de seguimiento de patrones por medio del procesamiento de imágenes que pueda aplicarse en cualquier ámbito, únicamente variando las características del objeto que se desea seguir, lo cual permitiría la aplicación de diversos campos donde se usen cámaras y sea necesario el seguimiento de alguna característica de un sujeto u objeto.

Para el desarrollo del sistema, se usará como herramienta de adquisición y procesamiento de imágenes el software de desarrollo matemático MatLab®, y la tarjeta de adquisición NI USB-6008-DAQ para entregar las señales de control al sistema de posicionamiento.

II. CONTENIDO

ASPECTOS GENERALES DEL PROCESO

En MatLab® se le llama objetos a los elementos que se crean para la adquisición de datos, ya sea de niveles de tensión o de imágenes (caso que va a ser de estudio en este documento), y por medio de estos objetos se le configuran los sistemas de adquisición; para el caso de la cámara usada en el proyecto relacionada a este artículo, se configurará la resolución (480 x 640) y el modelo de color de las imágenes, pues MatLab® está en la capacidad de adquirir imágenes en varios espacios de color, para este caso se trabajará con el más comúnmente utilizado, el RGB.

Las imágenes están compuestas por un arreglo de matrices que generan una matriz tridimensional con dimensiones de 480 en x, 640 en y, que definen el tamaño de la imagen (o también llamado resolución) y 3 (En Z) que corresponde a las

¹ Ingeniero Electrónico, Universidad Tecnológica de Pereira.

² Ingeniero Electricista, Especialista en Pedagogía para el desarrollo del aprendizaje autónomo, M. Sc. en Instrumentación Física.

componentes del modelo de color RGB. La primera matriz contiene información de las intensidades de los rojos (R), la segunda de los niveles de verde (G) y la tercera de los niveles de azul (B) presentes en la imagen.

En la figura 1 se muestra una imagen con alto contenido de color, las restantes imágenes en escala de gris, definen su correspondiente aporte en RGB, lo cual se aprecia al identificar que mientras disminuye la intensidad del gris la imagen aporta alto contenido de dicho color, por lo que las partes más intensas en gris indican menor aporte en la intensidad de color. Por ejemplo la guacamaya de color rojo (en la primera imagen) presenta un gris claro en la segunda imagen (Intensidad de color rojo) y grises más oscuros para las demás imágenes (Intensidades de color Verde y Azul). De otro lado al observar la guacamaya amarilla, se aprecia que para la construcción del amarillo se requiere alto contenido de rojo y verde, y una presencia casi nula de intensidad azul³.



Figura 1: Imagen a color y sus respectivas matrices RGB.

```
[filename pathname]=uigetfile...
    (*.jpg', 'Archivo de Imagen JPEG');
ruta=strcat(pathname,filename);
I=imread(ruta);
figure(1)
subplot(2,2,1)
imshow(I)
title('Imagen a Color')
subplot(2,2,2)
imshow(I(1:end,1:end,1))
title('Intensidad de color Rojo (R)')
subplot(2,2,3)
imshow(I(1:end,1:end,2))
title('Intensidad de color Verde (G)')
subplot(2,2,4)
imshow(I(1:end,1:end,3))
title('Intensidad de color Azul (B)')
```

Figura 2: Código para obtener la figura 1 (MatLab®).

³ Para entrar en mayores detalles sobre este análisis de contenido de color de una imagen, el lector puede implementar el código mostrado a continuación de la figura 1 (Código 1).

INICIALIZACIÓN DEL OBJETO DE ADQUISICIÓN DE IMAGEN

Al implementar el código de la figura 3, se obtiene la inicialización del objeto de video, por medio del cual se obtendrá la imagen para ser procesada, esto se realiza concurrentemente; es decir, se toma la imagen del objeto, y se procesa; cabe aclarar que dicho proceso es muy pesado (computacionalmente hablando) para MatLab®, por ende el video obtenido dará la sensación de estar retrasado, debido a lo anterior se considera que la mejor manera de implementar operaciones que involucren procesamiento de señales (imágenes, audio, ECG, ECG) o imágenes debe ser sobre un DSP⁴.

```
video=videoinput...
    ('winvideo',1,'YUY2_640x480');
set(video,'ReturnedColorSpace','rgb');
videoRes=get...
    (video,'VideoResolution');
nBands=get...
    (video,'NumberOfBands');
propimg=image(zeros(videoRes(2)...
    , videoRes(1), nBands));
preview(video,propimg)
```

Figura 3: Código inicialización del objeto de adquisición de imagen (MatLab®).

CAPTURA DE LA IMAGEN

Después de la creación del objeto (nombrado para este caso “video”), la cámara que se configuró⁵ se enciende automáticamente al correr el código, hasta este punto poniendo el video capturado en un objeto (*figure 1*), en esta parte de la configuración, la cámara ya está inicializada, el paso a seguir es realizar una captura de la imagen en determinado momento para procesarla, para tomar la imagen se implementa el código de la figura 4 que se muestra a continuación:

```
Img=getsnapshot(video);
```

Figura 4: Código para adquisición de una imagen del objeto “video” (MatLab®).

Al aplicar el código de la figura 4, se captura una imagen de la cámara y se almacena en la variable llamada *Img* en forma de una Matriz de [480,640,3], y es aquí donde ya es posible realizar un análisis a la imagen tomada, o en el caso que se desee, aplicarle el algoritmo de detección de patrones.

⁴ DSP: Digital Signal Processor (Procesador Digital de Señales).

⁵ En este caso YU_Y2 hace referencia a la cámara interna de un portátil, esto se puede identificar en la herramienta de adquisición de imágenes.

IDENTIFICACIÓN DE PATRONES

Los algoritmos de detección de patrones o eventos son muy comunes en la ingeniería, de hecho en su estado más íntimo, son filtros o selectores, estos pueden ser pasivo (Circuitos RC, RLC) o activos y programados (Sallen-Key, Butterworth, Tchebyshev, entre otros.), para citar algunos de los más usados y quizá conocidos se encuentran el algoritmo de detección de eventos de Pan-Tompkins (usado en señales electrocardiográficas), filtros de Kalman (usados desde la estimación de parámetros hasta procesamiento de imágenes).

Como se puede entender, la gran variedad de métodos de selección o identificación de características en algún campo específico es muy amplia y va desde procesos de selecciones simples como los de distancia euclidiana o vecinos más cercanos, hasta procesos mucho más elaborados y complejos como lógica difusa, redes neuronales, filtros de Kalman, etc. La diferencia más especial se relaciona con la carga computacional que cada proceso consume, ya que intrínsecamente y de manera independiente de la complejidad de sus sustentos teóricos, todos sirven para lo mismo, identificar y extraer características.

Para el caso relacionado en este trabajo, se recurre a la identificación del color en un proceso estadístico basado en la media y la desviación estándar. Para lo cual se creó un código en MatLab® que permite realizar la selección del segmento de imagen a la que se le quiere extraer las características de color, este proceso se puede realizar debido a que una imagen está compuesta por pixeles, y estos a su vez son interpretados por MatLab® como valores numéricos en un rango de 0 a 255 (del tipo entero sin signo de 8 bit's)⁶, donde cada valor hace referencia a un nivel de intensidad así como se mostró en la figura 1.

En la figura 5 se muestra el histograma⁷ de la componente de intensidades de color rojo de la figura 1.

Donde el nivel 0 (Negro) se interpreta como ausencia de color y 255 (Blanco) como intensidad total de color, de la figura 5 y teniendo en cuenta lo observado en la figura 1 en la intensidad de color rojo, se puede decir que la tendencia de la imagen "es a ser gris", lo cual se corrobora en el histograma de la figura 5, pues la mayoría de los pixeles de dicha imagen de encuentran entre 40 y 100 (niveles de intensidad de color).

⁶ Para lograr la aplicación de los algoritmos es necesario convertir los datos de entero sin signo (UINT8) a un tipo flotante (FLOAT), y posteriormente convertirlos al tipo UINT8 para poder ser entendido nuevamente como información correspondiente a una imagen, este es un error común en el procesamiento de imágenes con MatLab®.

⁷ Histograma: Contiene información sobre la frecuencia de los niveles de intensidad en una imagen en escala de grises.

SEGMENTACIÓN DEL OBJETO Y EXTRACCIÓN DE CARACTERÍSTICAS

El proceso de segmentar y extraer las características de color del objeto seleccionado se puede entender como un entrenamiento (el cual se incluirá en el algoritmo posteriormente desarrollado), para implementar este proceso primero se deben tomar muestras del objeto en condiciones semicontroladas de luz a fin de incluir suficientes características de color del objeto de prueba, lo cual le dará robustez al sistema para diferentes condiciones de trabajo.

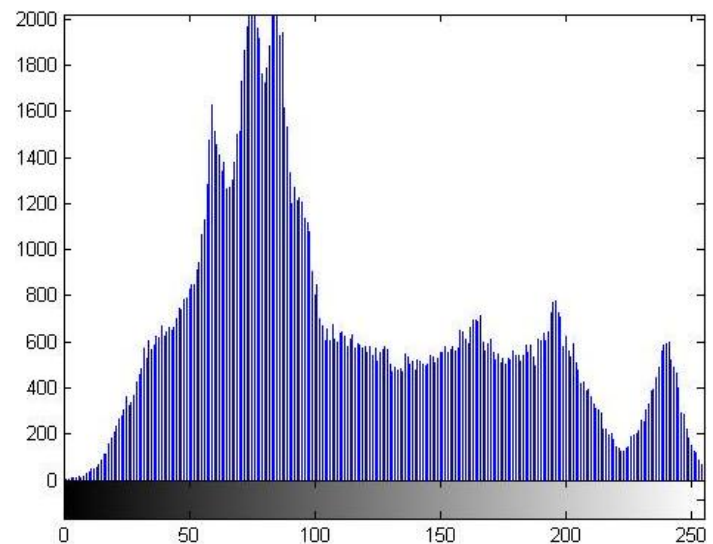


Figura 5: Histograma de la componente Roja de la figura 1.

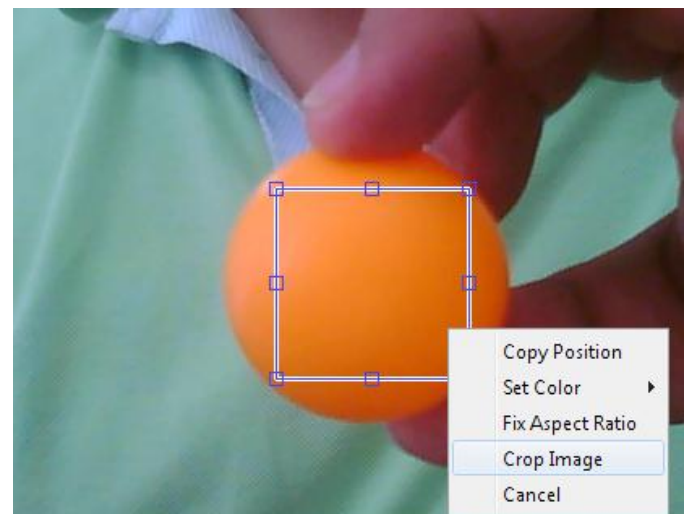


Figura 6: Realización de segmentación del objeto para extracción de características de color.

Basado en lo que se puede observar la figura 7, se logra estimar la tendencia de intensidad RGB para el objeto segmentado en la figura 6 y analizando estadísticamente el comportamiento de los pixeles que corresponden al objeto, y teniendo en cuenta que las condiciones de luz son medianamente controladas, se obtiene el resultado que se aprecia en la figura 8.

Los niveles medios se calcularon con la media de los datos tomados en la segmentación mostrada en la figura 6, y las cotas inferiores y superiores se calcularon restando y sumando respectivamente la desviación estándar de los datos, teniendo en cuenta que los datos son del tipo entero sin signo de 8 bit's (uint8). Este procedimiento se programó para ser realizado en las tres matrices de intensidad de color RGB.

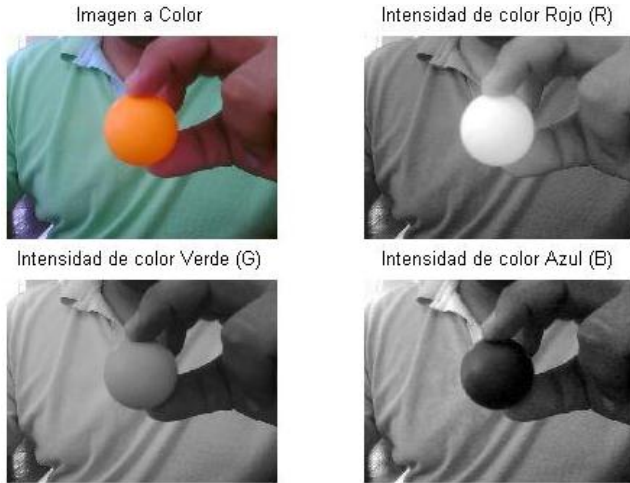


Figura 7: Imagen capturada y sus niveles de intensidad de RGB.

```

Command Window
-----
Límites para Rojo
ans =
    173    188    203

Límites para Verde
ans =
    84    107    129

Límites para Azul
ans =
     0     13     33
    
```

Figura 8: Resultado de los rangos de intensidad RBG del objeto.

APLICACIÓN DEL ENTRENAMIENTO

Para la aplicación del entrenamiento se recorren las matrices R G y B simultáneamente y se genera una matriz binaria donde se marcan los pixeles que cumplan con los rangos mostrados en la figura 8, en cuyo caso se estaría accediendo a un pixel que cumple con las características de color del objeto de prueba, en la figura 9A se muestra el

objeto de prueba y el mapa binario de la selección de los pixeles. Al tener completamente identificados los pixeles que contienen las características de color del objeto en el mapa binario, se realiza una extracción de características con funciones de MatLab®, donde se identifica el tamaño del objeto y su localización relativa en la imagen, debido a lo cual es posible encerrar el objeto como se muestra en la figura 9B, además estas características serán de utilidad para cerrar el lazo de control del sistema.

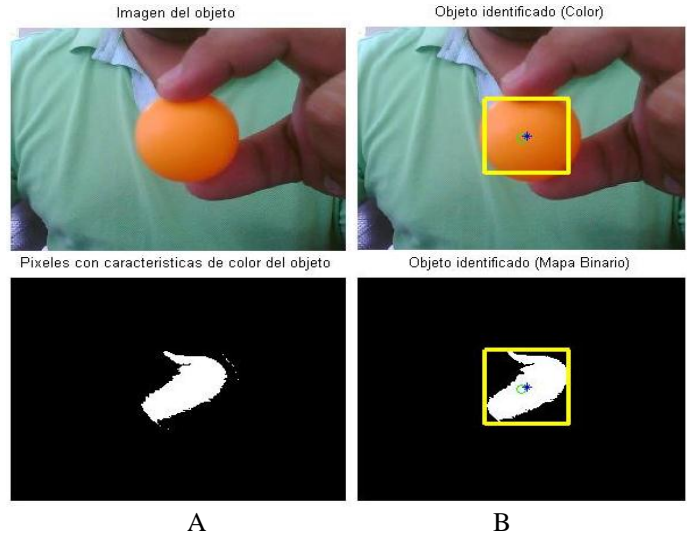


Figura 9: A) Mapa binario de identificación de pixeles del Objeto, B) Identificación del Objeto.

SISTEMA DE CONTROL

El sistema de control consta de dos motores que generan movimientos de vista horizontal y vertical, de cuya combinación se puede realizar el seguimiento del objeto. Desde el punto de vista del control se tiene un sistema de dos grados de libertad.

En este punto se puede realizar la siguiente pregunta: ¿Qué modelo de control implementar?: ¿Reubicación de polos?, ¿Control PID? ¿Realimentación de variables de estado?; a lo que se puede responder que al considerar que cualquiera es dependiente de un modelo del sistema físico, se aplicará inicialmente un sistema de estimación de parámetros de mínimos cuadrados a cada grado de libertad del sistema a fin de obtener el modelo de la planta para luego aplicar el modelo de control.

En la figura 10 se muestra el esquema del sistema físico de seguimiento a implementar, el cual se encargará del movimiento de la cámara para seguir el objeto.

El modelo de estimación de parámetros de mínimos cuadrados se muestra en la ecuación 1, este modelo de control es objeto de estudio para la aplicación al sistema de seguimiento, la viabilidad del uso de este método de estimación se debe a los

resultados obtenidos en las pruebas de simulación del algoritmo planteado en las ecuaciones 1 y 2.

La referencia del sistema de control siempre se relaciona con las coordenadas del centro de la imagen, a fin de lograr que en todo momento el sistema enfoque el objeto en el centro de la misma.

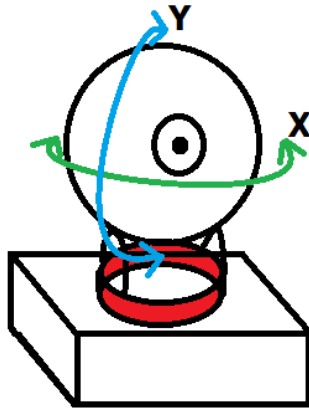


Figura 10: Sistema físico de movimiento de la cámara para el seguimiento de objeto.

$$\hat{\theta}_k = \hat{\theta}_{k-1} + \frac{P_{k-2}\varphi_{k-1}}{1 + \varphi_{k-1}^T P_{k-2} \varphi_{k-1}} (y_k - \varphi_{k-1}^T \hat{\theta}_{k-1}) \quad (1)$$

$$P_{k-1} = P_{k-2} - \frac{P_{k-2}\varphi_{k-1}\varphi_{k-1}^T P_{k-2}}{1 + \varphi_{k-1}^T P_{k-2} \varphi_{k-1}} \quad (2)$$

Dónde:

- $\hat{\theta}_k \rightarrow$ Vector de Parámetros de la planta estimada
- $y_k \rightarrow$ Muestras de la salida de la planta
- $\varphi_{k-1}^T \rightarrow$ Arreglo de las señales de entrada y salida
- $P_{k-1} \rightarrow$ Matriz de correlación

SIMULACIÓN DEL SISTEMA DE ESTIMACIÓN

Para probar el sistema de estimación se plantea la función de transferencia mostrada en (3):

$$H(s) = \frac{0.5}{s^2 + s + 1} \quad (3)$$

Discretizando la función de transferencia continua (3) con un tiempo de muestreo de 0,5 segundos se obtiene la función de transferencia discreta mostrada en (4).

$$H(z) = \frac{0.0522 z + 0.04414}{z^2 - 1.414 z + 0.6065} \quad (4)$$

Por tanto el sistema a estimar tiene una forma general como la que se muestra en (5).

$$H(z) = \frac{b_1 z + b_0}{z^2 + a_1 z + a_0} \quad (5)$$

Cuya ecuación en diferencias se muestra en (6):

$$y(k) = -a_1 y(k-1) - a_0 y(k-2) + b_1 u(k-1) + b_0 u(k-2) \quad (6)$$

Realizando una adecuación matemática, se puede obtener el modelo planteado en (6) en forma matricial convenientemente, por tanto en (7) se muestra en dicha forma:

$$y(k) = [-y(k-1) \quad -y(k-2) \quad u(k-1) \quad u(k-2)] \begin{bmatrix} a_1 \\ a_0 \\ b_1 \\ b_0 \end{bmatrix} \quad (7)$$

La escritura de la ecuación (7) se puede reescribir de manera reducida como se muestra en (8):

$$y(k) = \varphi_{k-1}^T \hat{\theta}_{k-1} \quad (8)$$

Dónde:

$$\varphi_{k-1}^T = [-y(k-1) \quad -y(k-2) \quad u(k-1) \quad u(k-2)] \quad (9)$$

$$\hat{\theta}_{k-1} = \begin{bmatrix} a_1 \\ a_0 \\ b_1 \\ b_0 \end{bmatrix} \quad (10)$$

Las expresiones (9) y (10) son requeridas para la aplicación del algoritmo de identificación de la función de transferencia, el siguiente dato necesario para la aplicación del algoritmo de mínimos cuadrados es conocer la matriz P, a esta matriz se le denomina matriz de correlación, la cual ayuda a la convergencia del algoritmo (ver referencia bibliográficas [3] y [4]).

El uso de estimadores de este tipo permite obtener funciones de transferencia de sistemas a los que no se les conocen dichos parámetros.

En la figura 11 se muestran sobre puestas las respuesta de la planta real (mostrada en (3)) y la planta estimada (mostrada en (11)), y en la figura 12 se muestra la evolución de los parámetros estimados.

En la figura 11 se observa que el estimador de parámetros converge aproximadamente a las 5 muestras, y el sistema estimado se obtiene utilizando los parámetros y la forma mostrada en (5), obteniendo la función de transferencia que se puede apreciar en la ecuación (11).

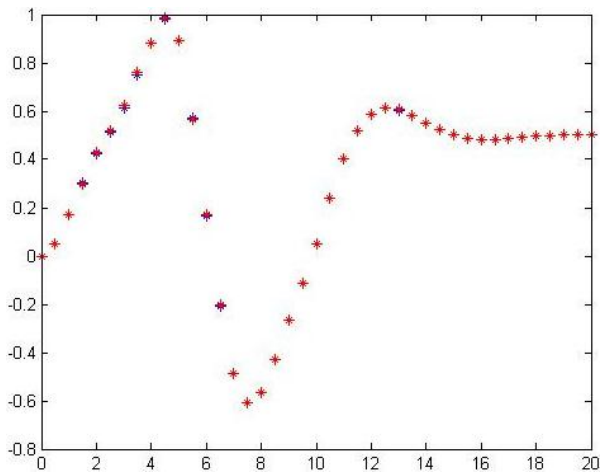


Figura 11: Respuesta real de la planta y respuesta estimada.

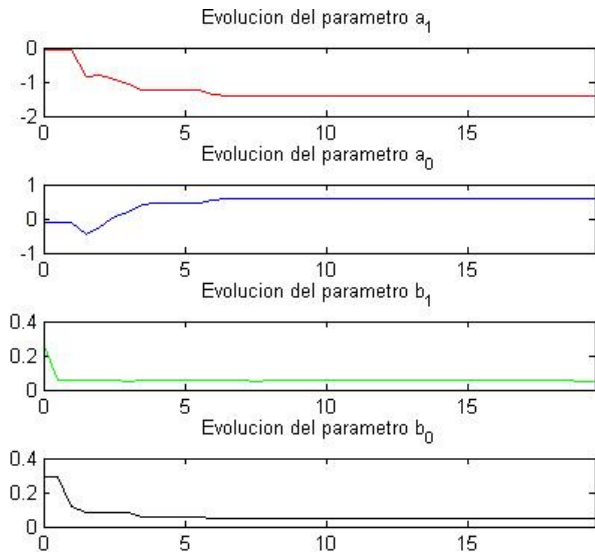


Figura 12: Evolución de los parámetros estimados.

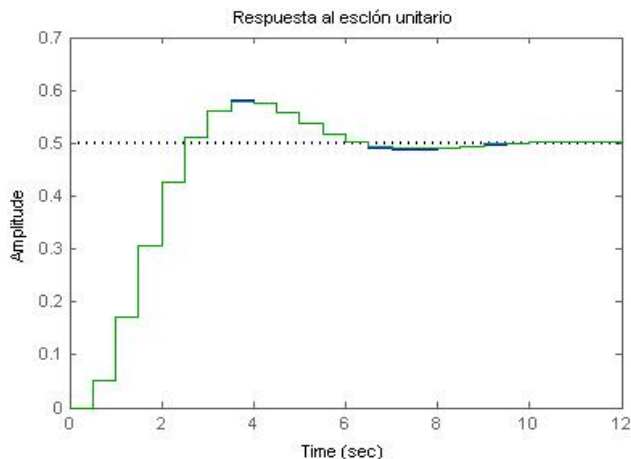


Figura 13: Respuesta al escalón unitario del sistema real discretizado (mostrado en (4)) y del sistema estimado (mostrado en (11)).

$$a_1 = -1.407 \quad a_0 = 0.6008 \quad b_1 = 0.05216 \quad b_0 = 0.04507$$

$$H(s) = \frac{0.05216 z + 0.04507}{z^2 - 1.407 z + 0.6008} \quad (11)$$

Con la obtención de la función de transferencia del sistema se puede aplicar cualquier técnica de control, cuya escogencia aún es objeto de estudio.

III. CONCLUSIONES

El prototipo planteado, implementa la aplicación de métodos de control y de procesamiento de imágenes, a fin de desarrollar un sistema sencillo de seguimiento de objetos móviles que tiene aplicación en diferentes contextos como sistemas de seguridad, filmación automática de eventos deportivos (con una cámara programada siguiendo un balón de fútbol a un jugador específico), entre otras.

Se evidencia por medio del desarrollo de este proyecto el potencial del procesamiento de imágenes como técnica de instrumentación en aplicaciones reales.

Las técnicas de estimación de parámetros permiten que el diseño de un sistema de control sea mucho más coherente y funcional en una aplicación real, pero como se evidencio, la aplicación de un método requiere de una capacidad de computo considerable que garantice el desempeño en tiempo real, por tanto, para la aplicación de este es necesario usar un sistema embebido que podría basarse en DSP's o FPGA's.

RECOMENDACIONES

En la parte de la aplicación del algoritmo de identificación del objeto, se pueden desarrollar algoritmos mucho más elaborados que identifiquen patrones más complejos que una característica de color (como es el caso de este trabajo), para ello, una extensión del proyecto es llevarlo a la aplicación de diferentes algoritmos para la identificación de patrones.

REFERENCIAS

- [1] Gonzalez R., & Woods R., "Digital image processing", 2ª edición, New Jersey, PRENTICE HALL, 2002, pág. 190.
- [2] González R., Woods R. & Eddis S., "Digital image processing using MatLab®", 2ª edición, New Jersey, PRENTICE HALL, 2003, pág. 595..
- [3] Ljung Lennart, "Sistema identificaion: Theory for the user", Suecia, PRENTICE HALL PTR, 1999, pág. 361.
- [4] Goodwin, "Adaptive Filtering, Prediction and Control", pag. 47.
- [5] Ogata Katsuhiko, "Ingeniería de control moderna", 3ª edición, Minnesota, PEARSON EDUCACION, 1998, pág.669 – 843.