

Memorias
XIII Simposio de Tratamiento
de Señales, Imágenes y Visión Artificial

STSIVA 2008

Tomo I

Bucaramanga, 10, 11 y 12 de Septiembre



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
B U C A R A M A N G A

DIRECTIVOS

Fray Orlando Rueda Acevedo, O.P.
Rector
Fray Guillermo León Villa Hincapié O.P.
Vicerrector Académico
Decano División de Ingenierías
Fray Guillermo León Villa Hincapié O.P.
Decano División de Ingenierías y Arquitectura
Fray José Rodrigo Arias Duque O.P.
Vicerrector Administrativo-Financiero

COMITÉ ORGANIZADOR

Luis Ómar Sarmiento Álvarez
Presidente Comité Organizador
Jesús E. Laborde
Coordinador Comité Organizador
Dolly Smith Flórez Moreno
Mónica Andrea Rico Martínez
Mayra Fernando Blanco Almeida
Ana Isabel Campos
Jenny Rocío Gómez
Jenniffer Dayanna Martínez
Adriana Quintana Ballén
María Lorenza Pinto Suárez
Jorge Alberto Martínez Campo

COMITÉ CIENTÍFICO:

Sonia Helena Contreras Ortiz
Juan Carlos Martínez Santos
Leonardo Ramírez
Pedro Raúl Vizcaya Guarín
Luis Fredy Sosa Quintero
Javier González
Mauricio Díaz
Alexánder Cerquera
Héctor Cancino
César Augusto Quintero Obando
Neil Guerrero González
Pablo Emilio Jojoa Gómez
Rodrigo Herrera
Edward Wilder Caro Anzola
Augusto Salazar
Diego Rolando Mahecha Capacho
Jhon Jairo Pérez Gelves
William Castro
Carlos Andrés Madrigal González
Fabián Jiménez López
Paola Fernanda Guzmán
Andrea Liliana Rodríguez Sánchez
Milton Forero López
César Augusto Isaza Bohórquez
Edwin Gómez
Rodolfo Sanchez García

Luis David Avendaño Valencia
Julio César García Álvarez
Jesús Vega
Julián Betancur

PRODUCCIÓN CREATIVA

Departamento de Publicaciones
Directora Dpto. Publicaciones
C.P. Luz Marina Manrique Cáceres
Diseño y Diagramación
Pub. Luis Alberto Barbosa Jaime

Impresión
DISTRIGRAF Impresores

© Universidad Santo Tomás

ISSN 978-958-8477-00-8

Contenido

Presentación	7
Caracterización de tejidos por medio del procesamiento digital de imágenes para el control de calidad del proceso de plastinación	9
Encriptación de huellas digitales utilizando transformada fraccional de Fourier	13
Modelado directo e inverso de imágenes infrarrojas de fase usando la teoría de cuadripolos térmicos en termografía de fase pulsada	23
Automatic analysis for the chemical testing of urine examination using digital image processing techniques	28
Optimización de la curva de distorsión utilizando cuantización vectorial de estado finito	32
Simulación de conversores A/D de alta resolución empleando modulación Delta-Sigma	36
Control visual basado en líneas utilizando un sistema de visión con dos cámaras	42
Aplicación de los sistemas de visión al control de movimiento de robots: Una revisión	48
Simulaciones para el aprendizaje de comunicaciones analógicas y digitales	52
Construcción y análisis no lineal de bases de datos de electrocardiogramas de felinos	58
Sistema de reconocimiento y lectura de placas de vehículos en movimiento	63
Evaluación de diferentes matrices de medición utilizando orthogonal matching pursuit (OMP)	67
Pathfinder: Estimación de la trayectoria de un vehículo a partir de una cámara embarcada	74

Estimación del fondo para la detección de vehículos en video	78
Metodología para evaluar el difeomorfismo de un atractor caótico usando el filtro de Kalman	83
Prototipo optoelectrónico basado en un correlador de transformada conjunta para verificación de huellas dactilares	87
Procesamiento digital de señales ultrasónicas TOFD JOURNALH	93
Codificación y compresión de señales de voz utilizando cuantización vectorial no determinística	98
Classification of one-dimensional motion from few spikes of a selected retinal ganglion cell population	102
Reconocimiento de objetos en una mesa X-Y	107
Modificador de tono de reproducción	112
Desarrollo de una herramienta software utilizando redes neuronales artificiales para estimar el comportamiento mecánico de arenas limpias pertenecientes a una cuenca Colombiana	116
Análisis de los sensores requeridos para el desarrollo de estimadores del comportamiento dinámico de la humedad en la membrana de una pila de combustible	121
Identificación de formas y posiciones de la boca en secuencias de imágenes faciales	126
Desarrollo del ejercicio de vigilancia tecnológica en arreglos análogos programables por campo (FPAA)	132
Implementación de una plataforma modular para monitoreo y caracterización de pilas de combustible - FCPUV	137
Tratamiento digital de señales electrooculográficas, aplicado al soporte en el diagnóstico de la esquizofrenia.	142

Estimación y dependencia de la evolución de parámetros de la máquina de inducción trifásica utilizando técnicas de procesamiento de señales	146
Reconocimiento de figuras en una mesa XY	150
Empleo de la transformada de fourier fraccional para extraer características discriminantes en el reconocimiento de locutores	156
Clasificación de movimientos de flexión y extensión en miembros superiores e inferiores con redes neuronales Feedforward	160
Análisis en tiempo frecuencia del ajuste de un automóvil a partir del sonido producido por el cierre de una de las puerta del vehículo	164
Sistema de medición de desplazamiento en dos ejes basado en técnicas de visión artificial	169
Sistema de reconstrucción de imágenes en 3D usando luz estructurada, implementado sobre una FPGA	173
Remoción del desplazamiento de la línea base en señales ECG aplicando filtrado de Kalman	177
Discriminación de eventos epilépticos mediante espectros paramétricos variantes en el tiempo	181
Diseño de sistema de adquisición óptico-digital para la inspección de la presencia de frutos de café en diferentes estados de maduración	185
Análisis en laboratorio del efecto de señales armónicas en motores de inducción mediante LABVIEW	189
Sistema SPRUA: Herramienta para el procesamiento y análisis de señales de audio	194
Minimización de artefactos fisiológicos en registros electroencefalográficos digitales de pacientes con epilepsias generalizadas	198
Optimización de la DTW basada en la naturaleza estadística de las disimilitudes	203

Reconstrucción tridimensional de la sección transversal de perfiles extruidos utilizando triangulación láser	207
Reconocimiento de figuras sobre una banda transportadora por medio de técnicas de visión e inteligencia artificial	211
Sistema de identificación de personas utilizando imágenes infrarrojas de la red vascular de la mano	216
Aplicación de la entropía en el análisis de señales respiratorias de pacientes en proceso de Weaning	220
Caracterización de la señal de flujo respiratorio mediante la estimación de instantes de inicio y fin de inspiración	224
Aplicación en DSP para modelamiento de sistemas dinámicos por medio de técnicas de filtrado adaptativo	228
Aplicación de la visión artificial en la determinación de los puntos de aprehensión de una prótesis de mano robótica	233
Detección de estructuras de información en señales de procesos complejos	237
Diseño e implementación de prácticas de laboratorio para el uso de DSP's 56800 de Motorola	242
Medición del espesor de la capa asfáltica usando visión artificial	247
Detección y seguimiento de vehículos automotores en video	251
Adquisición y procesamiento de señales en Labview con interfaz USB	255

Presentación

La División de Ingenierías de la Universidad Santo Tomás, Seccional Bucaramanga, tuvo el honor de ser el ente organizador del XIII Simposio de Tratamiento de Señales, Imágenes y Visión Artificial STSIVA 2008, durante los días 10, 11 y 12 de septiembre de 2008 en la ciudad de Bucaramanga, cuyo objetivo era el de promover el intercambio de experiencias académicas e investigativas en procesamiento de señales, imágenes y visión artificial en un espacio que permita la revisión del estado del arte y la actualización de métodos, técnicas y aplicaciones empleados en las diferentes áreas temáticas, tanto en el ámbito nacional como en el internacional.

El Simposio de Tratamiento de Señales, Imágenes y Visión Artificial, cuya sede se rota anualmente por las diferentes regiones del país, es un evento creado por la Sociedad Colombiana de Tratamiento de Señales y cuenta recientemente con el apoyo del Capítulo Colombiano de Procesamiento de Señales del IEEE. La realización de esta nueva versión contó adicionalmente con la cofinanciación de Colciencias y con la coordinación de la Línea de Investigación en Procesamiento de Señales perteneciente a la Unidad de Investigaciones de la Facultad de Ingeniería de Telecomunicaciones UNITEL, entre otras instancias de la Universidad Santo Tomás de Bucaramanga.

STSIVA 2008 rompió récord respecto a las versiones anteriores en cuanto a asistencia, número de trabajos inscritos y participación de ponentes internacionales, ya que contó con la inscripción de 365 participantes entre ponentes y asistentes, se recibieron 227 trabajos, de los cuales, después del proceso de evaluación por pares, se seleccionaron los mejores 107 artículos incluidos en su totalidad en la presente publicación, y se aceptaron en total seis artículos internacionales, dos de España, dos de Brasil, uno de Alemania y otro de Bélgica.

Las presentes memorias contienen los artículos de las ponencias realizadas en STSIVA 2008. La publicación se ha organizado en dos tomos, con 53 artículos el primero y 54 el segundo. El orden en el que se presentan los artículos responde únicamente al número consecutivo que se asignó a cada trabajo en el momento de la inscripción. Los trabajos se presentan sin hacer diferencia entre trabajos presentados en ponencias orales y los presentados como póster ya que la modalidad de los trabajos es independiente de la calidad de los mismos.

Los 107 artículos recibidos participaron en las siguientes temáticas: técnicas y aplicaciones del procesamiento de señales, procesamiento de imágenes y señales multidimensionales, tratamiento de señales de video, visión artificial, procesamiento de señales de origen biológico, aplicaciones desarrolladas con DSPS Y PLDS (FPGA), educación en señales, imágenes y visión artificial, tratamiento de señales de voz y audio y procesamiento de señales multimedia. Las temáticas con el mayor número de artículos aceptados fueron en su orden: técnicas y aplicaciones del procesamiento de señales, procesamiento de imágenes y señales multidimensionales, y visión artificial.

Paralelo al evento, se presentaron tres conferencias magistrales a cargo de expertos internacionales en procesamiento de señal. Los conferencistas y los temas fueron: El Dr. José Millet Roig de la Universidad Politécnica de Valencia participó con el tema “Aplicaciones del procesado digital de bioseñales a la cardiología”, el Dr. Alberto González Salvador, también de la Universidad Politécnica de Valencia con la conferencia “Algoritmos de detección por búsqueda en árbol (Sphere Decoders) para sistemas de comunicaciones MIMO”, y la Dra. Amy Reibman, conferencista distinguida del IEEE con el tema “Image and video quality assessment: The truth about PSNR”.

Como presidente del Comité Organizador de STSIVA 2008 y en representación de la Universidad Santo Tomás Seccional Bucaramanga, con la satisfacción de haber cumplido a con los objetivos propuestos y haber superado con creces las expectativas planteadas, con el agradecimiento a todas y todos quienes participaron en la organización del evento y en constancia del alto nivel académico y científico, pongo a su consideración esta edición impresa de las memorias del XIII Simposio de Tratamiento de Señales Imágenes y Visión Artificial.

Cordialmente,

Luis Ómar Sarmiento Álvarez
Presidente del Comité Organizador STSIVA 2008
losarmiento@ustabuca.edu.co

Caracterización de tejidos por medio del procesamiento digital de imágenes para el control de calidad del proceso de plastinación

José Arturo Lagos, Javier González Barajas.

Resumen—Categoría C4. Para tener en óptimas condiciones el material didáctico utilizado en asignaturas de anatomía y fisiología, se ha desarrollado la técnica de plastinación logrando la preservación de componentes de los sistemas orgánicos de seres vivos conservando, cualitativamente, su apariencia en color y textura. Esta técnica es de reciente creación y está basada en la sustitución de los fluidos de cada estructura por compuestos químicos basados en resinas y siliconas. Actualmente la evaluación de los resultados de este proceso de preservación, se realiza de forma cualitativa y se carece de técnicas cuantitativas para la evaluación de la calidad de la del proceso de plastinación. Este trabajo de investigación presenta la generación de una herramienta informática, que puede ser utilizada para la caracterización de los tejidos que hacen parte de las estructuras anatómicas sometidas al proceso de plastinación, y por medio del procesamiento digital de imágenes poder determinar cuantitativamente la calidad del proceso de plastinación.

Palabras claves— Plastinación, muestras anatómicas, procesamiento digital de imágenes.

I. INTRODUCCION

Dentro de la práctica de la anatomía se requiere la utilización de material proveniente de cadáveres con el fin de poder tener herramientas en la formación de estudiantes de área de las ciencias médicas y de la salud, lo cual requiere el uso de técnicas para la preparación y conservación de las estructuras orgánicas que provienen de los cuerpos utilizados en los anfiteatros. En primera instancia los cuerpos eran enterrados en la arena seca o entre las piedras de la región desértica, creando un ambiente de deshidratación y conservación ideales para este proceso [1,2,3,4]. En la actualidad se han introducido químicos como el formaldehído y las inyecciones arteriales [5]. La técnica de plastinación, desarrollada por Von Hagens, reemplaza los fluidos corporales de la muestra anatómica, por acetona en un proceso

de difusión. Seguidamente la acetona es sustituida por plásticos reactivos [6].

Dentro de las nuevas aplicaciones del tratamiento digital de imágenes, han surgido propuestas de soluciones para el estudio de tejidos, órganos y demás estructuras que hacen parte de los sistemas de un ser viviente [7]. Se han implementado soluciones para la detección de estructuras anómalas o características anormales de tejidos y también con el fin de extraer información de gran importancia en el análisis de patologías. Tomando como ejemplo el estudio de patologías cerebrales, se han implementado algoritmos de segmentación y operaciones morfológicas y además se han creado modificaciones del formato de representación de colores [8,9]. Se pueden citar otras aplicaciones orientadas a la detección de enfermedades de la sangre, por medio de análisis de células por medio de operaciones morfológicas [10]. Como también el desarrollo de métodos para la detección de células cancerígenas [11]. En el campo de la medicina veterinaria, se han utilizado técnicas de diagnóstico avanzado basado en procesamiento de imágenes, para el estudio de estructuras anatómicas de animales. Trabajos en los cuales se han tomado cortes transversales, preparados bajo la técnica de plastinación y han sido comparados con imágenes de tomógrafo estableciendo relaciones entre cortes de cada metodología de análisis de estructuras [12].

Como finalidad de este artículo se presenta la primera etapa de un proyecto que pretende lograr la implementación de sistemas basados en procesamiento digital de imágenes con el fin de efectuar controles de calidad en el proceso de plastinación, específicamente eliminando la subjetividad al evaluar, presentando valores estimados en contornos, tonalidades y morfología de los tejidos sometidos; además se entrega una medida cuantitativa de los cambios descritos, mostrando el porcentaje de diferencia y/o similitud entre dos o más procesos de conservación de tejidos. A continuación se presentan los resultados con respecto a la caracterización de tejidos sometidos al proceso de plastinación comparados con muestras tratadas con métodos tradicionales de conservación.

Artículo presentado al evento Simposio Nacional de Tratamiento de Señales, Imágenes y Visión Artificial. Celebrado en la Universidad Santo Tomás, Bucaramanga Colombia. 2008.

Javier González Barajas. Facultad de Ing. Electrónica y Biomédica. Universidad Antonio Nariño. Cra 3 Este N° 47ª-15. Bogotá. Colombia. (e-mail: javiere_gonzalez@yahoo.com.mx).

José Arturo Lagos Sandoval. Facultad de Ing. Electrónica y Biomédica. Universidad Antonio Nariño. (e-mail: josearturo@yaho.es).

II. MATERIALES Y MÉTODOS

Teniendo en cuenta que el proceso de preparación de tejidos basados en la técnica de plastinación, tiene como principal objetivo la preservación de estas estructuras manteniendo todas sus características, se propone en este proyecto la utilización de técnicas del procesamiento digital de imágenes para el desarrollo de una aplicación orientado a la caracterización de los tejidos sometidos al proceso de plastinación. Utilizando las diversas estrategias matemáticas que se han utilizado para la caracterización de tejidos, con el fin de extraer información de los tejidos sometidos al proceso de plastinación que sea útil para la evaluación de la calidad de este proceso de conservación.

El desarrollo de este proyecto de investigación se basa en el seguimiento diferentes etapas. El proyecto empieza con la recolección de muestras de tejidos de estructuras anatómicas antes y después del proceso de plastinación. Las pruebas preliminares se realizaron con corazones de caninos. Seguidamente se procedió a la adquisición de imágenes, por medio de una cámara digital marca Kodak de 10 mega píxeles de resolución. Posteriormente se procede a filtrar, suavizar y caracterizar las imágenes con el programa Eikona 3D y algunos aportes de software libre como Image Java.

III. RESULTADOS

Una vez realizado el filtrado y preprocesamiento necesario para tener la imagen sin ruidos e interferencias y con un histograma ecualizado, se procede a extraer características de los tejidos que conforman las diversas imágenes de prueba realizando prueba de caracterización con dos imágenes patrón una de corazón en estructura completa y la segunda en un corte sagital del mismo, obteniendo como resultado una parametrización de la imagen en píxeles y luego comparar con el mismo órgano sumergido en cloroformo y otros compuestos destinados a conservar el tejido. Ver figura 1.

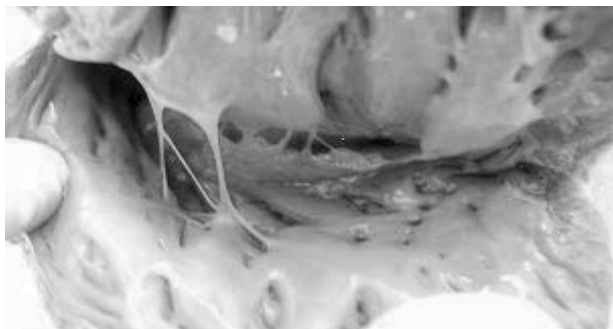


Fig. 1. Imagen adquirida de un corte corazón de canino en condiciones naturales.

La figura 2 ilustra el histograma de las cada una de las componentes RGB de una sección de la imagen de la figura 1 seleccionada aleatoriamente.

La figura 3 ilustra una imagen tomada de un corazón de canino almacenado en formol por espacio de dos semanas.

Nuevamente tomando una porción aleatoriamente de la

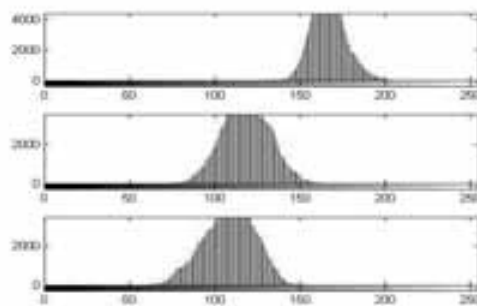


Fig. 2. Histograma de componentes R, G y B respectivamente de la imagen adquirida de un corte corazón de canino en condiciones naturales.

imagen adquirida se procede a extraer el histograma de cada una de las componentes R, G y B. La figura 4 ilustra el resultado de la extracción del histograma de la imagen de un corazón de canino almacenado en formol.



Fig. 3. Imagen adquirida de un corazón de canino almacenado en formol durante dos semanas.

Con el fin de lograr otro referente de comparación, se procedió a tomar corazones de caninos y almacenarlos en nevera por espacio superior a dos semanas.

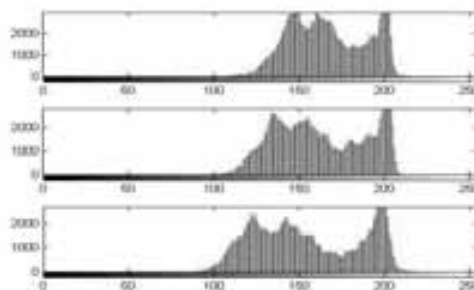


Fig. 4. Histograma de componentes R, G y B respectivamente de la imagen adquirida de un corte corazón de canino almacenado en formol.

La figura 5 ilustra la imagen adquirida de un corazón de canino almacenado en nevera durante dos semanas.

Para obtener otro referente de comparación, se procedió a tomar corazones de caninos y almacenarlos en nevera por espacio superior a dos semanas.

Tomando una muestra aleatoriamente de la figura 5, se extrae el histograma ilustrado en la figura 6.



Fig. 5. Imagen adquirida de un corazón de canino almacenado en nevera durante dos semanas

El resultado de los histogramas ofrece información acerca de cómo se pueden alterar las distribuciones de las componentes R, G y B de la forma anatómica en estado natural cuando se somete a un método de conservación utilizando formol o el almacenamiento a bajas temperaturas en neveras.

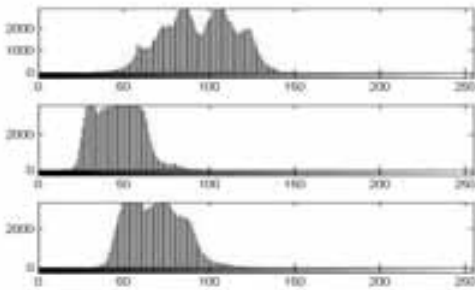


Fig. 6. Histograma de componentes R, G y B respectivamente de la imagen adquirida de un corte corazón de canino almacenado en nevera.

El preprocesamiento preparó la imagen para su caracterización y permitió determinar con un alto grado de precisión los componentes de color, intensidad y morfología estructural de los tejidos para realizar la comparación con los tejidos plastinados



Fig. 7. Binarización de la imagen adquirida de la muestra de corazón de canino en estado natural.

El paso posterior a la extracción de histogramas fue la binarización y extracción de contornos de las imágenes adquiridas. Las figura 7 ilustra el resultado del proceso de binarización aplicado a la muestra contenida en la figura 1

La figura 8 contiene el resultado del proceso de extracción de contornos aplicado a la muestra de corazón de canino en estado natural.



Fig. 8. Extracción de contornos de la imagen adquirida de la muestra de corazón de canino en estado natural.

Este ensayo del proceso nos muestra que la similitud de morfología y tonalidad se mantiene en un 70 % en imágenes plastinadas con respecto a la imagen natural, mientras que en imágenes conservadas en cloroformo o en neveras la similitud es inferior al 45 % en la mayoría de los casos.

Lo anterior da pie para afirmar que una vez realizado el proceso completo con más de 30 imágenes patrón y comparadas con el mismo número de imágenes plastinadas, la posibilidad de conservar tejidos por este método es bastante cercana al tejido original (en la mayoría de sus características), sin embargo, esto no lo podemos decir con suficiente certeza aun, solo cuando tengamos el proceso completo dentro de ocho meses, como se planteó en el proyecto.

IV. CONCLUSIONES

El procesamiento digital de imágenes aplicado al proceso de plastinación nos permite caracterizar y medir diferentes parámetros (tonalidad, morfología, degradación, etc) de un tejido sometido a dicho proceso, generando una forma cuantitativa para determinar la calidad de la plastinación, y excluyendo la subjetividad en el proceso de evaluación que se hace hoy día sobre la calidad del proceso.

La comparación de imágenes a través de su caracterización digital nos permite conocer el grado de diferencia o similitud entre dos imágenes de un mismo tejido, controlando las condiciones de iluminación bajo la cual fueron tomadas.

El procesamiento digital de imágenes es un desarrollo que no solo evita la subjetividad al evaluar imágenes, sino que también nos permite automatizar procesos que se realizaban exclusivamente en forma manual y precaria.

Los resultados presentados permiten hacer comparaciones entre una muestra anatómica en su estado natural y métodos tradicionales de conservación que en este caso son el almacenamiento en formol y el sometimiento a bajas temperaturas en nevera.

REFERENCIAS

- [1] El hombre, origen y misterios. México, Uteha, 1983. Vol. 2, p. 1443.
- [2] El hombre, origen y misterios. México, Uteha, 1983. Vol. 2, p. 193.
- [3] Goerke H. 300 Años de Historia de la medicina. Barcelona, Gustavo Gilli, 1986. p. 11.
- [4] Palencia Oyarzábal C. La medicina en la historia. México, Médico Moderno, 1996. Vol. 1, p. 84.
- [5] Reyes-Aguilar María E. Anatomía Humana y Plastinación. Bol Mex His Fil Med 2007; 10(1): 24-39.
- [6] Von Hagens G. Bodyworlds. 3a. ed. Heidelberg, Institut für Plastination, 2002. p. 19 - 25.
- [7] Carr Joseph, Brown John. Introduction to Biomedical equipmen Technology. Prentice Hall. USA. 1998. 702: p604-620.
- [8] Semmlow John. Biosignal and Biomedical Image Processing. Matlab based Aplication. Marcel Dekker. USA. 2004. 443: p271-301.
- [9] Martínez R. Manuel. Detección de Astrocitoma Cerebral Infantil Aplicando Segmentación de Imágenes y Morfología Matemática. Rev Mex Neuroci 2006; 7(5): 400- 404.
- [10] Di Ruberto, C., Dempster, A., Khan, S., Jarra, B., (2002), Analysis of infected blood cell images using morphological operators, Image and Vision Computing, 20 122-146.
- [11] Mogilner, J., Eldar, S., Sabo, E., Hassoun, M.,Attias, D.(2000), Computer-assisted image analysis can aid the prognostication of advanced-stage neuroblastomas, Cancer Res. Clin. Oncol, 126:285-90.
- [12] Hörst König, González Hernán. Secciones Anatómicas de la Cabeza del Perro. Consideraciones Prácticas en Relación con los Métodos Modernos de Diagnóstico por Imágenes. Agro-Ciencia 17(I): 37- 41, 2001.
- [13] Bedoya, J., Branch, J. y Restrepo, A. (2002), Image analysis to characterize tsp in medellin, in '95th Annual Conference and Exhibition The Air & Waste Management Association's 2002 Annual Conference & Exhibition Proceedings on June 23-27', Baltimore, Maryland, USA. ISBN: 0-923204-45-8.
- [14] Branch, J. y Olague, G. (2001), 'La visión por computador: Una aproximación al estado del arte', Revista Dyna (133). ISSN 0012-7353.
- [15] Worjnar, L. (1999), Image Analysis. Applications in Materials Engineering, CRC Press, Boca Raton.
- [16] N Montes, "Desarrollo de algoritmos de segmentación de frutos maduros y verdes de café en imágenes tomadas en condiciones controladas, basados en las propiedades de color", Tesis Doctoral, Universidad Nacional de Colombia, 2001.

Encriptación de huellas digitales utilizando transformada fraccional de Fourier

Juan M. Vilardy, Yaileth J. Morales, Cesar O. Torres, Lorenzo. Mattos.

Resumen— Un nuevo método de encriptación de imágenes utilizando La Transformada Fraccional de Fourier Discreta (DFrFT) y La Transformada Jigsaw es propuesto. La principal característica de este esquema de encriptación es que la imagen encriptada es real y tiene el mismo tamaño que la imagen original, siendo esta imagen encriptada mucho más conveniente para su almacenamiento o transmisión por redes de comunicación digital. El uso de La Transformada Jigsaw incrementa la seguridad de la imagen encriptada y elimina las mascararas de fases aleatorias utilizadas en muchos criptosistemas de imágenes basados en Transformada Fraccional de Fourier (FrFT). Por último para el nuevo método de encriptación propuesto, fue desarrollado un algoritmo digital de encriptación de huellas digitales. En el algoritmo criptográfico implementado doce llaves son usadas, constituidas por los seis órdenes fraccionales de las DFrFTs y las seis permutaciones aleatorias utilizadas en las transformadas Jigsaw, todas esas llaves son necesarias para una correcta desencriptación, obteniendo un alto nivel de seguridad en la protección de las huellas digitales para una determinada aplicación.

Palabras Claves— Biometría, Encriptación/Desencriptación de Imágenes, Huellas Digitales, Transformada Fraccional de Fourier Continua/Discreta, Transformada Jigsaw, Permutaciones Aleatorias, Error Cuadrático Medio (MSE), Relación Señal a Ruido (SNR).

I. INTRODUCCIÓN

La seguridad de la información esta siendo más y más importante con el rápido desarrollo de la internet y las telecomunicaciones en la ultima década y el precio a pagar por la seguridad de la información digital es fuertemente encarecido. El cifrado de imágenes es la forma más directa de proteger estos datos digitales, debido a la redundancia y la correlación de las imágenes. El cifrado de imágenes es diferente de la criptología tradicional y acerca de la información de la imagen, la seguridad del estudio tradicional

Este trabajo fue desarrollado en el Laboratorio de Óptica e Informática de La Universidad Popular Del Cesar (Valledupar, Colombia).

Juan M. Vilardy. Ingeniero Electrónico, Joven Innovador de COLCIENCIAS adjunto al Grupo de Óptica e Informática. (e-mail: vilardy.juan@unicesar.edu.co).

Yaileth J. Morales. Ingeniera Electrónica, Joven Innovadora de COLCIENCIAS adjunto al Grupo de Óptica e Informática. (e-mail: yaileth12@hotmail.com).

Cesar O. Torres. Dr. en Física. Docente asociado a La Universidad Popular Del Cesar, Director del Grupo de Óptica e Informática. (e-mail: torres.cesar@caramail.com).

Lorenzo. Mattos. Msc. en Informática. Docente asociado a La Universidad Popular Del Cesar, Coordinador del Grupo de Óptica e Informática. (e-mail: mattos.lorenzo@caramail.com).

de criptología no es buena.

Muchos sistemas de encriptación son basados en la transformada de Fourier (FT) o la FrFT [1-10], estos sistemas comúnmente emplean mascararas de fases aleatorias puras, tales como la doble codificación de fase aleatoria [3-4], para adicionar ruido sobre la información de la imagen. Uno de los principales inconvenientes de tales esquemas de encriptación es que la imagen encriptada es compleja y por lo tanto esto seria inadecuado para aplicaciones reales.

El encriptamiento de huellas digitales (imágenes) que utiliza DFrFT y transformada Jigsaw surge como la aplicación y la generalización de los trabajos realizados de cifrado de imágenes digitales mediante la FrFT [11] y FT [12-13], respectivamente, debido a que la DFrFT es una extensión de la Transformada de Fourier Discreta (DFT), agregando así nuevas características que aumentan los niveles de seguridad en las huellas digitales encriptadas, para permitir así una mayor protección a las bases de datos que contienen este tipo de imágenes (huellas digitales).

El informe de resultados sobre la investigación de encriptación de huellas digitales vía DFrFT y transformada Jigsaw esta organizado de la siguiente manera: En la sección II se presenta la FrFT/DFrFT y sus propiedades. En la sección III se tiene la definición de la transformada Jigsaw. La formulación matemática de la técnica de encriptación y desencriptación es descrita en la sección IV. La sección V muestra en detalle el algoritmo digital para el proceso de encriptación/desencriptación de las imágenes (huellas digitales). Los resultados del algoritmo diseñado e implementado son mostrados en la sección VI. La seguridad del algoritmo digital propuesto es analizada en la sección VII. Finalmente, las conclusiones son hechas en la sección VIII.

II. FRFT/DFRFT Y PROPIEDADES IMPORTANTES

La FrFT es una generalización de la transformada identidad y la FT en dominios fraccionales [14]. La tradicional FT descompone señales en sinusoides mientras la FrFT expresa las señales en términos de una base ortonormal formada por chirps. La FrFT Continua puede ser entendida como una FT a la p^{th} potencia, donde p no es requerido que sea entero. La FrFT de orden p , es un operador lineal integral que mapea una función dada $f(x)$ a una función $f_p(\xi)$, así [15]:

$$f_p(\xi) = \mathfrak{F}^p[f(x)] = \int_{-\infty}^{+\infty} K_p(\xi, x) f(x) dx \quad (1)$$

Donde el kernel esta definido por:

$$K_p(\xi, x) = C_\alpha \exp\left\{-i\pi\left(2\frac{x\xi}{\sin\alpha} - (x^2 + \xi^2) \cot\alpha\right)\right\} \quad (2)$$

Con:

$$C_\alpha = \sqrt{I - i \cot\alpha} = \frac{\exp\{-i[(\pi \operatorname{sgn}(\sin\alpha)/4) - \alpha/2]\}}{\sqrt{|\sin\alpha|}} \quad (3)$$

$$\alpha = \frac{\pi p}{2} \quad (4)$$

Observando que α , al definirse como un número real, solo aparece como argumento de funciones trigonométricas en las ecuaciones (2) y (3), la definición (1) es periódica en p , con periodo 4.

La DFrFT que se utiliza en esta investigación es la definida por Candan [16], la cual es en una dimensión:

$$f_p[n] = F^p(f[n]) = \sum_{n=0}^N F^p[m, n] f[n] \quad (5)$$

El kernel es:

$$F^p[m, n] = \sum_{k=0, k \neq (N-1+(N)_2)}^N u_k[m] e^{-i\frac{\pi}{2}kp} u_k[n] \quad (6)$$

Donde $u_k[n]$ es la k -ésima función de Hermite Gauss discreta y $(N)_2 = N \bmod 2$, esta transformación discreta también es periódica en p , con periodo 4.

Las propiedades más importantes de la FrFT/DFrFT son:

1) *Aditividad del orden fraccional o índice:*

$$\mathfrak{F}^{p_1} \{\mathfrak{F}^{p_2}[f(x)]\} = \mathfrak{F}^{p_1+p_2}[f(x)] \quad (7)$$

2) *Unicidad:*

$$\{\mathfrak{F}^p[f(x)]\}^* = \mathfrak{F}^{-p}[f^*(x)] \quad (8)$$

Donde: * denota el complejo conjugado.

3) *Reducción a la transformada de Fourier (continua o discreta) cuando $p=1$.*

III. TRANSFORMADA JIGSAW

La transformada Jigsaw, $J\{\cdot\}$, se define como una juxtaposición de diferentes secciones de una imagen [8]. Un simple caso bidimensional es mostrado en la Fig. 1.

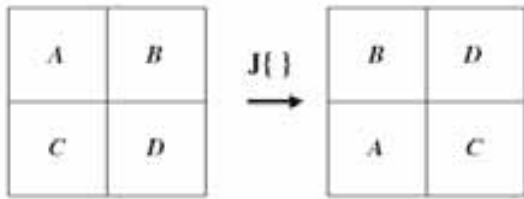


Fig. 1. Simple ilustración de la transformada Jigsaw.

El efecto de esta transformada sobre una imagen (Fig. 2a) es mostrado en la Fig. 2b.

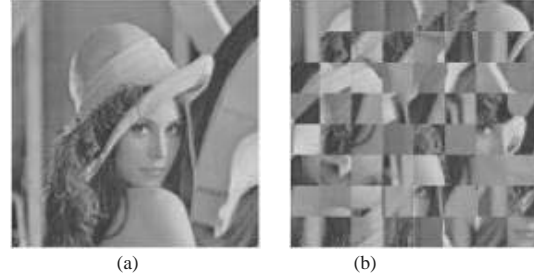


Fig. 2. (a) Imagen de entrada de 256x256 píxeles, (b) Imagen de entrada después de las primeras 16x16 transformadas Jigsaw.

En este caso la imagen fue dividida en 64 subsecciones de 8x8 píxeles, las cuales son reposicionadas relativamente de acuerdo a alguna permutación. La permutación usada es aleatoria. La transformada Jigsaw es unitaria, es decir que la energía se conserva al aplicar esta transformada y esta también tiene inversa. En el caso mostrado en la Fig. 2b hay 64! posibles permutaciones para la transformada Jigsaw. Cada particular transformada Jigsaw es denotada por algún índice, es decir: $J_b\{\cdot\}$ y su inversa es denotada por $J_{-b}\{\cdot\}$.

IV. FORMULACIÓN MATEMÁTICA DEL MÉTODO DE ENCRIPTAMIENTO Y DESENCRIPTAMIENTO

Sea $I(x, y)$ una imagen real a encriptar de $M \times N$ píxeles (M y N , número de filas y columnas respectivamente, en escala de grises) e $I_c(z, y)$ una imagen compleja de $M/2 \times N$ píxeles:

$$I_c(z, y) = I_R(z, y) + jI_I(z, y), \quad z = 0, 1, \dots, \frac{M}{2} - 1; y = 0, 1, \dots, N - 1 \quad (9)$$

Donde: $I_R(z, y)$ es la imagen parte real de $I_c(z, y)$ construida de la primera mitad de filas y todas las columnas de $I(x, y)$ y $I_I(z, y)$ es la imagen parte imaginaria de $I_c(z, y)$ construida de la segunda mitad de filas y todas las columnas de $I(x, y)$. La imagen compleja $I_c(z, y)$ contiene toda la información de la imagen a encriptar en su parte real e imaginaria. El proceso de encriptación se le aplica a la imagen compleja $I_c(z, y)$, transformando esta imagen doce veces con Jigsaw y FrFT, las siguientes ecuaciones describen el proceso de encriptación:

$$I_f(u, v) = \mathfrak{F}^y \left\{ J_{Rb3} \left\{ \mathfrak{F}^b \left\{ J_{Ib2} \left\{ \mathfrak{F}^a \left\{ J_{Rb1} \left\{ I_c(z, y) \right\} \right\} \right\} \right\} \right\} \right\} \quad (10)$$

$$I_E(r, s) = \mathfrak{F}^x \left\{ J_{Ib6} \left\{ \mathfrak{F}^z \left\{ J_{Rb5} \left\{ \mathfrak{F}^\sigma \left\{ J_{Ib4} \left\{ I_f(u, v) \right\} \right\} \right\} \right\} \right\} \right\} \quad (11)$$

$$I_E(r, s) = I_{ER}(r, s) + jI_{EI}(r, s), \quad r = 0, 1, \dots, \frac{M}{2} - 1; s = 0, 1, \dots, N - 1 \quad (12)$$

$$I_e(p, s) = \begin{bmatrix} I_{ER}(r, s) \\ I_{EI}(r, s) \end{bmatrix}, \quad p = 0, 1, \dots, \frac{M}{2} - 1; s = 0, 1, \dots, N - 1 \quad (13)$$

$$\begin{aligned} \text{si, } g(n, q) &= f(n, q) + jh(n, q) \\ &\Rightarrow \begin{cases} J_{Rb}\{g(n, q)\} = J_b\{f(n, q)\} + jh(n, q) \\ J_{Ib}\{g(n, q)\} = f(n, q) + jJ_b\{h(n, q)\} \end{cases} \end{aligned} \quad (14)$$

Donde: $I_E(r, s)$ es la imagen encriptada compleja de $M/2 \times N$ píxeles ($I_f(u, v)$ es una imagen compleja intermedia del proceso de encriptación), $I_e(p, s)$ es la imagen encriptada puramente real de $M \times N$ píxeles, construida de la parte real e imaginaria de la imagen encriptada compleja, como lo muestra la

ecuación (13). J_{Rb} y J_{Ib} son transformadas Jigsaw de índice b , que actúan sobre la parte real e imaginaria respectivamente de una imagen compleja, $\alpha, \beta, \gamma, \sigma, \chi$ y ε son los ordenes fraccionales utilizados.

Para el procedimiento de descriptación se aplica el procedimiento de encriptación en el sentido inverso, con los índices negativos de las transformadas Jigsaw y los ordenes fraccionales de la FrFT también negativos, utilizando la propiedad de aditividad de la FrFT y la transformada inversa de Jigsaw:

$$I_e(p, s) = \begin{bmatrix} I_{ER}(r, s) \\ I_{EI}(r, s) \end{bmatrix}, p = 0, 1, \dots, M-1; s = 0, 1, \dots, N-1 \quad (15)$$

$$I_E(r, s) = I_{ER}(r, s) + jI_{EI}(r, s), r = 0, 1, \dots, \frac{M}{2}-1; s = 0, 1, \dots, N-1 \quad (16)$$

$$I_f(u, v) = J_{-Ib4} \left\{ \mathfrak{S}^{-\sigma} \left\{ J_{-Rb5} \left\{ \mathfrak{S}^{-\chi} \left\{ J_{-Ib6} \left\{ \mathfrak{S}^{-\varepsilon} \left\{ I_E(r, s) \right\} \right\} \right\} \right\} \right\} \right\} \quad (17)$$

$$I_D(z, y) = J_{-Rb1} \left\{ \mathfrak{S}^{-\alpha} \left\{ J_{-Ib2} \left\{ \mathfrak{S}^{-\beta} \left\{ J_{-Rb3} \left\{ \mathfrak{S}^{-\gamma} \left\{ I_f(u, v) \right\} \right\} \right\} \right\} \right\} \right\} \quad (18)$$

$$I_D(z, y) = I_c(z, y) = I_R(z, y) + jI_I(z, y) \quad (19)$$

$$\begin{aligned} \text{si, } g(n, q) &= f(n, q) + jh(n, q) \\ \Rightarrow \begin{cases} J_{-Rb} \{g(n, q)\} = J_{-b} \{f(n, q)\} + jh(n, q) \\ J_{-Ib} \{g(n, q)\} = f(n, q) + jJ_{-b} \{h(n, q)\} \end{cases} \end{aligned} \quad (20)$$

Ahora $I_D(z, y)$ es la imagen descriptada compleja. Y así por ultimo para recuperar la imagen real que inicialmente fue encriptada, a partir de la ecuación (19), se tiene:

$$I(x, y) = \begin{bmatrix} I_R(z, y) \\ I_I(z, y) \end{bmatrix}, x = 0, 1, \dots, M-1; y = 0, 1, \dots, N-1 \quad (21)$$

V. ALGORITMO DIGITAL PARA LA ENCRIPCIÓN DE IMÁGENES

La implementación del algoritmo digital de encriptación se realizó en la plataforma de programación Matlab[®] v.7.5, debido a sus grandes facilidades y su alto rendimiento en el manejo de matrices (imágenes).

Las imágenes digitales a encriptar son huellas digitales con las siguientes características: formato *.bmp en escala de grises, 144x144 píxeles ($M=N=144$), 8 bits de cuantización (los valores de los píxeles son números enteros entre 0 y 255) y 24 KB de tamaño en disco.

A. Algoritmo Digital de Encriptación

1) Conversión de digital a análogo de la imagen de $M \times N$ píxeles a encriptar. Los valores de los píxeles de la imagen digital son escalados a valores decimales entre cero y uno.

2) Construcción de la imagen compleja $I_c[z, y]$ de $M/2 \times N$ píxeles a partir de la imagen real a encriptar (la imagen resultante del paso uno), como lo indica la ecuación (9).

3) Generación de los seis ordenes fraccionales de las DFrFTs (números decimales mayores que cero y menores que

cuatro) y las seis permutaciones aleatorias empleadas en las transformadas Jigsaw (dichas permutaciones aleatorias son vectores de 144 posiciones que contienen los números enteros del 1 hasta el 144, ubicados en posiciones aleatorias para cada vector) a partir de una clave alfanumérica (código ASCII) que podrá tener máximo diez caracteres.

4) Aplicación de la transformada Jigsaw J_{Rb1} , para esto se divide la imagen parte real de $I_c[z, y]$ en 72 subsecciones de 12×12 píxeles y cada subsección es reposicionada aplicando J_{b1} con la primera permutación aleatoria generada en el paso tres, la imagen parte imaginaria de $I_c[z, y]$ no es modificada.

5) Aplicación de la DFrFT en una dimensión de orden α (con un kernel de $M/2$ filas y $M/2$ columnas, para las huellas digitales bajo estudio el kernel es de 72×72 píxeles) a la imagen compleja resultante del paso cuatro. La DFrFT que se aplica en esta investigación a la imagen compleja de $M/2$ filas y N columnas, es la siguiente multiplicación matricial:

$$I_\alpha[m, n] = K[m, n]I_{c1}[m, n] \quad (22)$$

Donde $K[m, n]$ es el kernel de la transformación discreta de orden α en una dimensión de $M/2$ filas y $M/2$ columnas, $I_{c1}[m, n]$ es la imagen compleja a transformar de $M/2$ filas y N columnas y $I_\alpha[m, n]$ es la imagen transformada con orden α , que también tiene $M/2$ filas y N columnas; Se decidió aplicar la DFrFT en esta forma debido a que el algoritmo utilizado para calcular dicha transformada [17], toma un tiempo computacional considerable cuando se requiere aplicar transformada en dos dimensiones (es decir transformar primero las filas y luego las columnas), [18], y solo se necesita que el algoritmo empleado cumpla con la propiedad de aditividad del orden fraccional de la DFrFT, para la aplicación específica de este trabajo de investigación.

6) El resultado del paso anterior se le aplica transformada Jigsaw J_{Ib2} , para esto se divide la imagen parte imaginaria de $I_\alpha[m, n]$ en 72 subsecciones de 12×12 píxeles y cada subsección es reposicionada aplicando J_{b2} con la segunda permutación aleatoria generada en el paso tres, la imagen parte real de $I_\alpha[m, n]$ no es modificada. Posteriormente a este resultado se aplica DFrFT de orden β .

7) Se aplica transformada Jigsaw J_{Rb3} al resultado del paso seis (con la tercera permutación aleatoria generada en el paso tres, de la misma forma que se aplico J_{Rb1} en el paso cuatro) y dicho resultado se le aplica DFrFT de orden γ .

8) La imagen digital resultante del paso siete se le aplica transformada Jigsaw J_{Ib4} (con la cuarta permutación aleatoria generada en el paso tres, de la misma forma que se aplico J_{Ib2} en el paso seis) y posteriormente se aplica DFrFT de orden σ .

9) Aplicación de la transformada Jigsaw J_{Rb5} al resultado del paso ocho (con la quinta permutación aleatoria generada en el paso tres, de la misma forma que se aplico J_{Rb1} en el paso cuatro) y a este resultado se le aplica DFrFT de orden χ .

10) Aplicación de la transformada Jigsaw J_{Ib6} al resultado del paso nueve (con la sexta permutación aleatoria generada en el paso tres, de la misma forma que se aplico J_{Ib2} en el paso seis) y a dicho resultado se le aplica DFrFT de orden ε , la imagen resultante de este paso es la imagen encriptada compleja de $M/2 \times N$ píxeles.

11) Como el resultado del paso diez es una matriz de números complejos, se separa en dos matrices diferentes la parte real e imaginaria de dicha matriz compleja.

12) Se obtienen los números menores de las matrices parte real e imaginaria del paso anterior, si estos números son negativos, entonces estos números son restados a cada matriz respectivamente, para así obtener solo valores positivos en estas matrices; si estos números menores son positivos no se ejecuta ninguna operación y los números menores son colocados a ceros, en caso contrario serán guardados (como números negativos).

13) Luego de lograr solo números positivos en las matrices parte real e imaginaria del paso once, se normalizan dichas matrices, guardando en un archivo de texto (*.txt) los números mayores de cada matriz y los números menores encontrados en el inciso doce.

14) Por último las dos matrices parte real e imaginaria de la imagen encriptada compleja luego de ser restadas y normalizadas en los pasos doce y trece respectivamente, conforman la imagen encriptada puramente real de $M \times N$ píxeles, donde la primera mitad de filas y todas las columnas de la imagen encriptada real esta conformada por la matriz parte real de la imagen encriptada compleja de $M/2 \times N$ píxeles y la segunda mitad de filas y todas las columnas de la imagen encriptada real esta conformada por la matriz parte imaginaria de la imagen encriptada compleja de $M/2 \times N$ píxeles, como lo muestra la ecuación (13). La imagen encriptada puramente real es guardada como un archivo de imagen en cualquier formato convencional (*.jpeg, *.jpg, *.bmp, *.png).

Los ordenes fraccionales de las DFrFTs aplicadas están restringidos para valores reales mayores que cero y menores que cuatro, debido a la periodicidad de la DFrFT y también dado que los ordenes cero y cuatro dan como resultado la misma entrada cuando se aplica una FrFT ó DFrFT. Las permutaciones aleatorias empleadas son vectores de longitud igual a 144, por que las subsecciones de las imágenes tomadas para aplicar transformada Jigsaw tienen 144 píxeles.

B. Algoritmo Digital de Descriptación

El proceso de descriptación es el mismo proceso de encriptación, pero en el sentido inverso con los índices negativos de las transformadas Jigsaw y los ordenes fraccionales de la DFrFT también negativos, y por último se ordenan las matrices resultantes de la imagen descriptada compleja (ecuación (19)) en la forma indicada por la ecuación (21), para así obtener la imagen real que inicialmente fue cifrada de $M \times N$ píxeles. En la Fig. 3, se muestra el diagrama de bloques del proceso de encriptación de la imagen digital.

VI. RESULTADOS EXPERIMENTALES

Antes de examinar los resultados del proceso de encriptación sobre las huellas digitales, en la Fig. 4a se muestra la huella digital de prueba y en la Fig. 4b se tiene el efecto de la transformada Jigsaw sobre la huella digital de la Fig. 4a.

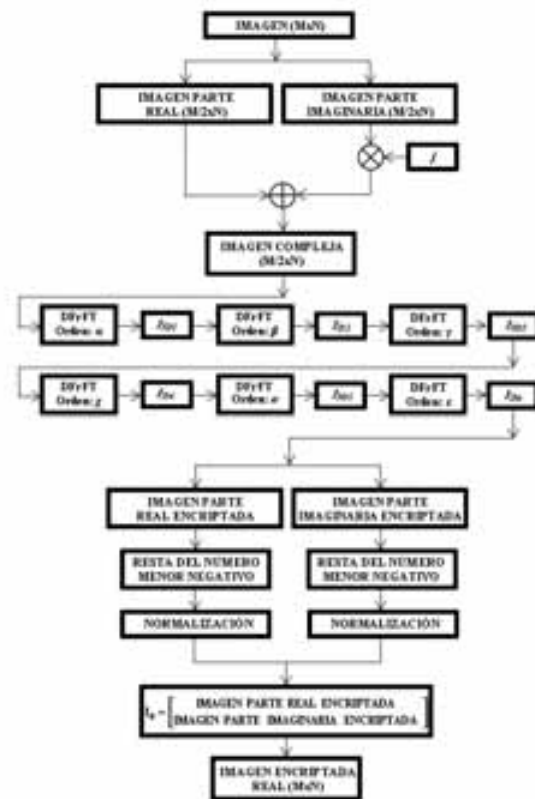


Fig. 3. Diagrama de bloques del proceso de encriptación de la imagen digital.

Al realizar el proceso de encriptación de la huella digital de $M \times N$ píxeles mediante el algoritmo descrito en la sección V, se consigue que la huella digital encriptada (imagen real de $M \times N$ píxeles) oculte la totalidad de la información contenida, como se aprecia en la Fig. 4c, la distribución de intensidades de la huella digital encriptada varía al cambiar las llaves (ordenes fraccionales de las DFrFT y las permutaciones aleatorias empleadas en las transformadas Jigsaw, es decir la clave alfanumérica). Cuando se realiza el proceso de descriptación con las llaves correctas (clave alfanumérica correcta), se logra recuperar la huella digital original con pérdidas no visibles al ojo humano, como lo muestra la Fig. 4d.

Si las llaves utilizadas en el proceso de descriptación no son iguales a las llaves utilizadas en el proceso de encriptación, la huella digital no se recuperará, como se aprecia en la Fig. 4e. De la anterior figura se observa que con cambios pequeños en las llaves correctas, aparece mucha distorsión en la huella digital recuperada. Es obvio que el proceso de descriptación es más sensible al orden fraccional ϵ que α , dado que un pequeño error ocurrido en la etapa de aplicación de la DFrFT de orden ϵ en la descriptación será acumulado y eventualmente destruiría la recuperación de la huella digital, mientras la dependencia en la descriptación sobre el orden fraccional α es ligeramente menos insensible. Y por último si las permutaciones aleatorias

de las transformadas Jigsaw utilizadas en el proceso de descryptación no son iguales a las utilizadas en el proceso de encryptación, la huella digital no se recuperará, así como lo muestra la Fig. 4e. El efecto de cambiar los órdenes fraccionales y las permutaciones aleatorias sobre la huella digital recuperada es observado en la Fig. 4e, por lo tanto si los ordenes fraccionales cambian, entonces las permutaciones aleatorias también cambiarán y esto se debe a la modificación de la clave alfanumérica (ya que de dicha clave alfanumérica se generan las doce llaves de los algoritmos digitales de encryptación/descryptación).

La huella digital encryptada fue almacenada en el disco duro del PC con los siguientes formatos de imágenes empleando 8 bits de cuantización, en la Tabla 1 se muestra el formato utilizado y el respectivo tamaño en disco que ocupa cada formato, cabe recordar que la huella digital a encryptar tiene las siguientes características: formato *.bmp en escala de grises, 144x144 píxeles, 8 bits de cuantización y 24 KB de tamaño en disco.

TABLA 1. COMPARACIÓN DEL TAMAÑO EN DISCO OCUPADO POR LOS FORMATOS CONVENCIONALES DE IMÁGENES UTILIZADOS PARA ALMACENAR LA HUELLA DIGITAL ENCRYPTADA.

Formato	*.png	*.bmp	*.jpeg	*.jpg
Tamaño en disco (KB)	20	24	20.2	20.2

De la anterior tabla se tiene que la huella digital encryptada almacenada con formato *.png es la que menos ocupa tamaño en el disco duro del PC y es 4 KB menor que la huella digital a encryptar, por este resultado la huella digital encryptada en este trabajo será almacenada con formato *.png. A continuación en la siguiente tabla se tiene el tamaño en disco que ocupa la huella digital encryptada al variar el Nivel de Cuantización (NC) con formato *.png:

TABLA 2. TAMAÑO EN DISCO DE LA HUELLA DIGITAL ALMACENADA PARA DIFERENTES NIVELES DE CUANTIZACIÓN CON FORMATO *.PNG.

NC	8	9	10	11	12	13	14	15	16
Tamaño en disco (KB)	20	28	32	36	36	36	40	40	44

Al observar la Tabla 2, es obvio que al aumentar los niveles de cuantización de la huella digital encryptada cuando esta es almacenada, se tiene que el tamaño en el disco duro del PC de dicha huella digital encryptada también aumenta hasta por lo menos el doble del tamaño en disco de la huella digital a encryptar, cuando se utilizan 16 bits de cuantización.

El Error Cuadrático Medio (MSE, Mean Square Error) y la Relación Señal a Ruido (SNR, Signal-to-Noise Ratio) entre la huella digital de entrada y nuestra huella digital descryptada son calculados para validar la confiabilidad del algoritmo implementado. El MSE puede ser definido por la diferencia de energía entre las huellas digitales de entrada y descryptada,

así:

$$MSE = \frac{1}{M \cdot N} \sum_{x=1}^M \sum_{y=1}^N [I[x, y] - I_1[x, y]]^2 \quad (23)$$

Donde $I[x, y]$ y $I_1[x, y]$ son las matrices elemento de la huella digital de entrada y nuestra huella digital descryptada en el píxel $[x, y]$, respectivamente, y $M \cdot N$ es el tamaño de la huella digital. *El MSE entre la huella digital original y la huella descryptada correctamente que se muestra en la Fig. 2d es 0, y el correspondiente MSE entre la Fig. 2e y la huella digital original es 17503.4508.*

La SNR es definida como, [19]:

$$SNR = \frac{\sum_{x=1}^M \sum_{y=1}^N I[x, y]}{\sum_{x=1}^M \sum_{y=1}^N [I[x, y] - I_1[x, y]]^2} \quad (24)$$

La SNR entre la huella digital original y la huella descryptada correctamente que se muestra en la Fig. 2d es ∞ , y la correspondiente SNR entre la Fig. 2e y la huella digital original es 1.1945.

A continuación se presenta los resultados sobre el estudio del MSE y la SNR para cuando los niveles de cuantización (en bits) son cambiados al guardar las huellas digitales encryptadas y luego son descryptadas. La huella digital a encryptar esta en la Fig. 4a, las huellas digitales descryptadas para las huellas digitales encryptadas con distintos niveles de cuantización son mostradas en la Fig. 5. Los niveles de cuantización utilizados al guardar las huellas digitales encryptadas influyen sobre el MSE y la SNR de la siguiente forma, ver Tabla 3.

Al incrementar el nivel de cuantización que se emplea para almacenar la huella digital encryptada, se asegura un resultado óptimo en el proceso de descryptación, obteniendo un MSE muy bajo y una SNR muy elevada; *No obstante con un nivel de cuantización de 8 bits para la huella digital encryptada, se obtiene una huella digital descryptada con un MSE muy pequeño de 1.1218 (comparado con el máximo valor, el cual es 65536) y una SNR de 18638.732, y visualmente la imagen descryptada se muestra perfectamente definida al igual que la huella digital de prueba (ver Fig. 5a).*

La sensibilidad de los ordenes fraccionales de las DFrFTs son examinados introduciendo errores en estos (de manera individual) y dejando fijas las permutaciones aleatorias utilizadas en las transformadas Jigsaw. El MSE y la SNR son empleados para medir el nivel de protección sobre las huellas digitales encryptadas. En esta prueba de desviación de los ordenes fraccionales de las DFrFTs sobre los valores correctos del proceso de descryptación, se introduce un error que varía entre -2 y 2, posteriormente para cada variación se calcula el

MSE y la SNR, los resultados se observan en la Fig. 6.

De experimentos computacionales se encontró que los ordenes fraccionales son sensible a una variación de $1e-4$, al presentar un MSE mayor y una SNR mas baja que los obtenidos con 8 bits de cuantización (ver Tabla 3). La discontinuidad en la curva de la SNR en las proximidades y en el punto cero (ninguna desviación sobre los ordenes fraccionales, Fig. 6b) quiere decir que el valor de la SNR es demasiado grande o infinito.

En la siguiente prueba, se chequea la tolerancia a la pérdida de la información de la huella digital encriptada cuando esta es desencriptada, para esta prueba se realiza la oclusión del 25% y 50% de los píxeles de la huella digital encriptada. La Fig. 7 muestra las huellas digitales ocluidas y las correspondientes huellas digitales recuperadas con su respectivo MSE y SNR, las huellas digitales de las Fig. 7a y Fig. 7k son las huellas digitales encriptada y recuperada, respectivamente, y así sucesivamente hasta el par de imágenes de las Fig. 7j y Fig. 7t. Para analizar los resultados de la Fig. 7, es necesario tener presente que la imagen (huella digital) encriptada puramente real de $M \times N$ píxeles esta conformada por una imagen parte real e imaginaria y estas dos ultimas imágenes constituyen la imagen encriptada compleja de $M/2 \times N$ píxeles, es decir que la primera mitad de filas y todas las columnas de la imagen encriptada puramente real es la imagen parte real de la imagen encriptada compleja (es decir el 50% de píxeles superior de la imagen encriptada puramente real) y la segunda mitad de filas y todas las columnas de la imagen encriptada puramente real es la imagen parte imaginaria de la imagen encriptada compleja (es decir el 50% de píxeles inferior de la imagen encriptada puramente real).

Por lo tanto al ocluir la imagen parte real o imaginaria de la huella digital encriptada compleja, es decir el 50% píxeles superior o inferior de la imagen encriptada puramente real como se ve en las Fig. 7b y Fig. 7d, respectivamente, las imágenes originalmente encriptadas no se podrán recuperar, ver Fig. 7l y Fig. 7n, respectivamente, igual resultado se tiene para las huellas digitales encriptadas de las Fig. 7i y Fig. 7j y sus correspondientes huellas digitales desencriptadas, Fig. 7s y Fig. 7t, respectivamente. Ahora cuando se ocluye el 25% de píxeles de la huella digital encriptada puramente real, Fig. 7a, Fig. 7c, Fig. 7e y Fig. 7f, solamente se recupera la mitad de la huella digital que se encriptó, ver Fig. 7k, Fig. 7m, Fig. 7o y Fig. 7p, esto se debe a que se elimina la mitad de información almacenada en las imágenes parte real o imaginaria de la imagen encriptada compleja; De igual manera para las huellas digitales encriptadas de las Fig. 7g y Fig. 7h y sus correspondientes huellas digitales desencriptadas, Fig. 7q y Fig. 7r, respectivamente, se tiene el mismo resultado anterior, debido a que se elimina la mitad de información almacenada en las imágenes parte real e imaginaria de la imagen encriptada compleja. **De esta prueba se puede decir que el algoritmo digital de encriptación/desencriptación propuesto**

tiene una correspondencia de píxeles uno a uno entre la imagen compleja a encriptar y la imagen compleja desencriptada, y así de esta manera se modifican los píxeles de la imagen desencriptada puramente real, en concordancia a lo explicado en la sección V.

Para la implementación del algoritmo digital en Matlab[®] v.7.5, se utilizó un computador IBM Pentium 4 con un procesador de 2.23 GHz y 256 MB de RAM, obteniendo los siguientes tiempos de encriptación/desencriptación para una huella digital de 144x144 píxeles:

Tiempo de Encriptación: 0.083786 Segundos.

Tiempo de Desencriptación: 0.082066 Segundos.

VII. SEGURIDAD DEL ALGORITMO DIGITAL PROPUESTO

A. Espacio de Clave o Llave

Para el análisis de espacio de clave del algoritmo implementado, solamente se consideran los caracteres imprimibles del código ASCII (Los códigos del 32 al 256 en decimal, se conocen como caracteres imprimibles, y representan el carácter espacio, letras, dígitos, signos de puntuación y varios símbolos), ahora como la clave alfanumérica utilizada en el algoritmo de encriptación tiene 10 caracteres ASCII, para tener éxito en un ataque de fuerza bruta sobre la huella digital encriptada hay 225^{10} posibilidades para la clave alfanumérica. **Para la huella digital de prueba utilizada en este trabajo, el tiempo de desencriptamiento es 0.082066 segundos y por lo tanto para tener éxito en un ataque de fuerza bruta se tendrían que emplear $2.728905e22$ segundos, esto es equivalente a $8.653301e14$ años.**

B. Ataques Estadísticos

Para mostrar que la huellas encriptadas con el esquema propuesto en este trabajo resisten ataques estadísticos, se utiliza la trasformada discreta de Fourier bidimensional (2D-DFT), [20]. El espectro de la huella digital de prueba se encuentra en la Fig. 8 y el espectro de la huella digital encriptada se presenta en la Fig. 9.

De la Fig. 8, se observa que la distribución de frecuencias de la huella digital de prueba está concentrada en una pequeña área, la cual sufre el riesgo de interceptación de información, mientras que la distribución de frecuencias de la huella digital encriptada (Fig. 9) se ha aplanado. Por lo tanto, se ha validado que la huella digital de prueba está ocultada perfectamente contra ataques estadísticos con el algoritmo criptográfico propuesto en este trabajo de investigación.

C. Comentario Sobre un Ataque Específico de los Sistemas de Encriptación basados en FrFT [22]

En investigaciones recientes los sistemas de encriptación basados en La Transformada Fraccional de Fourier Discreta de Múltiples-Parámetros (The Multiple-Parameter Discrete Fractional Fourier Transform, MPDFRFT, [21]) y operaciones

lineales son considerados inseguros [22]. *Debido a que la DFrFT es un caso particular de la MPDFRFT, el ataque propuesto en la referencia 22 podría violar la seguridad del algoritmo digital de encriptación propuesto en este trabajo, pero las huellas digitales encriptadas en este trabajo resisten este ataque específico, por que las transformadas Jigsaw aplicadas en el algoritmo de encriptación son operaciones no lineales de reorganización de píxeles que emplean permutaciones aleatorias de longitud igual a 144 (vectores), y por lo tanto se tienen 144! posibilidades para la permutación aleatoria a emplear.*

VIII. CONCLUSIÓN

Un nuevo esquema de encriptación de imágenes utilizando la DFrFT y la transformada Jigsaw ha sido presentado. La principal ventaja del esquema de encriptación presentado radica en que la imagen encriptada es puramente real y tiene el mismo tamaño de la imagen (también puramente real) a encriptar. El uso de la DFrFT en la encriptación de imágenes aumenta enormemente los parámetros de seguridad de la imagen encriptada, debido a la sensibilidad en cualquier cambio que se haga sobre los órdenes fraccionales utilizados (clave alfanumérica), y adicional a esto las seis permutaciones aleatorias utilizadas en las transformadas Jigsaw incrementan mucho más la seguridad para cualquier criptoanalista que intente desencriptar la imagen digital sin ser autorizado. Un algoritmo digital de encriptación/desencriptación para huellas digitales fue desarrollado e implementado, encontrándose que el nivel de cuantización utilizado para almacenar la huella digital encriptada influye sobre el proceso de desencriptamiento (MSE y SNR), ya que si se utiliza más de 8 bits para guardar la huella digital encriptada se obtendrán mejores resultados en las huellas digitales recuperadas (ver Tabla 3), pero la huella digital encriptada ocupará un mayor tamaño de almacenamiento en el disco duro del PC, como lo muestra la Tabla 2; Para usos posteriores de las huellas digitales desencriptadas (como pueden ser procesos de reconocimiento o autenticación dactilar) hay que tener presente los resultados del MSE y la SNR obtenidos para cada nivel de cuantización (Tabla 3) y determinar de que manera estos parámetros (MSE y SNR) influyen sobre la huella digital desencriptada para dicho proceso posterior. El algoritmo digital implementado tiene una correspondencia de píxeles uno a uno entre la imagen compleja a encriptar y la imagen compleja desencriptada, adicionalmente las huellas digitales encriptadas resisten los ataques de fuerza bruta, estadísticos, y el ataque específico mencionado en la referencia 22. Finalmente, para incrementar la velocidad de los procesos de encriptación/desencriptación de las huellas digitales, se debe trabajar sobre la obtención de algoritmos rápidos y eficientes para calcular la DFrFT, ya que en la actualidad no se tienen resultados satisfactorios en esta área.

AGRADECIMIENTOS

Este trabajo de investigación fue soportado por la

Universidad Popular del Cesar y COLCIENCIAS – Instituto Colombiano para el Desarrollo de La Ciencia y La Tecnología “Francisco José de Caldas”, bajo el programa de: Jóvenes Innovadores e Investigadores de COLCIENCIAS.

REFERENCIAS

- [1] N. K. Nishchal, G. Unnikrishnan, J. Joseph, K. Singh, Fully phase encryption using fractional Fourier transform, *Opt. Eng.* 42, 1583-1588, (2003).
- [2] N. Singh, A. Sinha, Optical image encryption using fractional Fourier transform and chaos, *Opt. Lasers Eng.* 46, 117-123, (2008).
- [3] N. K. Nishchal, G. Unnikrishnan, J. Joseph, K. Singh, Fully phase encryption using fractional Fourier transform, *Opt. Eng.* 42, 1583-1588, (2003).
- [4] P. Refregier, B. Javidi, Optical image encryption based on input plane and Fourier plane random encoding, *Opt. Lett.* 20, 767-769, (1995).
- [5] G. Unnikrishnan, J. Joseph, K. Singh, Optical encryption by double-random phase encoding in the fractional Fourier domain, *Opt. Lett.* 25, 887-889, (2000).
- [6] B. Hennelly, J. T. Sheridan, Fractional Fourier transform-based image encryption: phase retrieval algorithm, *Opt. Comm.* 226, 61-80, (2003).
- [7] B. Zhu, S. Liu and Q. Ran, Optical image encryption based on multifractional Fourier transforms, *Opt. Lett.* 25, 1159-1161, (2000).
- [8] B. Hennelly, J. T. Sheridan, Optical image encryption by random shifting in fractional Fourier domains, *Opt. Lett.* 28, 269-271, (2003).
- [9] S. Liu, L. Yu, B. Zhu, Optical image encryption by cascaded fractional Fourier transforms with random phase filtering, *Opt. Comm.* 187, 57-63, (2001).
- [10] Z. Liu, M. A. Ahmad, S. Liu, Image encryption scheme based on the commutation and anti-commutation rules, *Opt. Comm.* 279, 285-290, (2007).
- [11] J. Vilardy, J. Calderon, C. Torres, and L. Mattos, The Digital Images Phase Encryption Using Fractional Fourier Transform, In: *Proceedings of the Electronics, Robotics and Automotive Mechanics Conference, IEEE*, ISBN: 0-7695-2569-5/06, 2006.
- [12] L. Brito, C. Torres, Encriptamiento de imágenes digitales a color mediante transformada de Fourier, In: *Revista Colombiana de Física, VOL. 35, No.1, 2003.*
- [13] L. Brito, C. Torres, Empleo de tres llaves en la encriptación de imágenes digitales a color mediante transformada de Fourier, In: *X Simposio de Tratamiento de Señales, Imágenes y Visión Artificial, Cali – Colombia, 2005.*
- [14] M. A. Kutay, H. M. Ozaktas, O. Arikan, L. Onural, Optimal Filtering in Fractional Fourier Domains, *IEEE Transactions on Signal Processing, VOL. 45, No.5, MAY 1997.*
- [15] H. Ozaktas, Z. Zalevsky, M. A. Kutay, *The Fractional Fourier Transform with Applications in Optics and Signal Processing*, Jhon Wiley & Sons (eds). New York, 2001.
- [16] C. Candan, M. A. Kutay, H. M. Ozaktas, The Discrete Fractional Fourier Transform, *IEEE Transactions on Signal Processing, VOL. 48, No.5, MAY 2000.*
- [17] C. Candan, dFRT: The Discrete Fractional Fourier Transform, A Matlab Program, 1998, <http://www.ee.bilkent.edu.tr/~haldun/dFRT.m>.
- [18] A. Bultheel, H. Martínez, Computation of the fractional Fourier transform, *ELSEIVER, Applied and Computational Harmonic Analysis, 16 (2004) 182-202.*
- [19] T. Nomura, B. Javidi, Optical encryption system with a binary key code. *Appl Opt* 2000; 39(26):4783-7.
- [20] F. Han, J. Hu, X. Yu, Y. Wang, Fingerprint images encryption via multi-scroll chaotic attractors, *ELSEIVER, Applied Mathematics and Computation, 185 (2007) 931-939.*
- [21] S.-C. Pei, and W.-L. Hsue, The Multiple-Parameter Discrete Fractional Fourier Transform, *IEEE Signal Processing Letters, VOL. 13, NO. 6, June 2006.*
- [22] Amr M. Youssef, On The Security of a Cryptosystem Based on Multiple-Parameter Discrete Fractional Fourier Transform, *IEEE Signal Processing Letters, VOL. 15, 2008.*

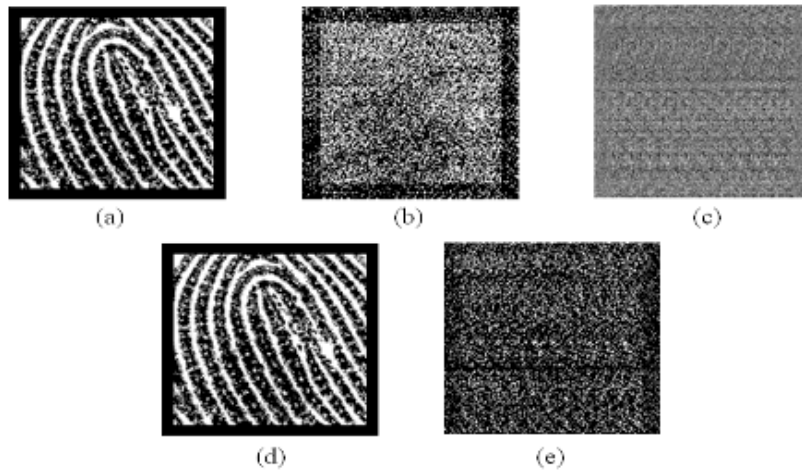


Fig. 4. (a) Huella digital de prueba, (b) Efecto de la transformada Jigsaw sobre la huella digital de prueba, (c) Huella digital encriptada con la clave alfanumérica: 'vilardy121' ($\alpha=1.005$, $\beta=1.41$, $\gamma=1.7124$, $\sigma=1.0819$, $\chi=1.054$ y $\varepsilon=1.189$), (d) Huella digital descryptada con la llave alfanumérica correcta, (e) Huella digital descryptada con un error en la llave alfanumérica: (e) 'vilardy120' ($\alpha=1.005$, $\beta=1.41$, $\gamma=1.7124$, $\sigma=1.0818$, $\chi=1.054$ y $\varepsilon=1.189$).

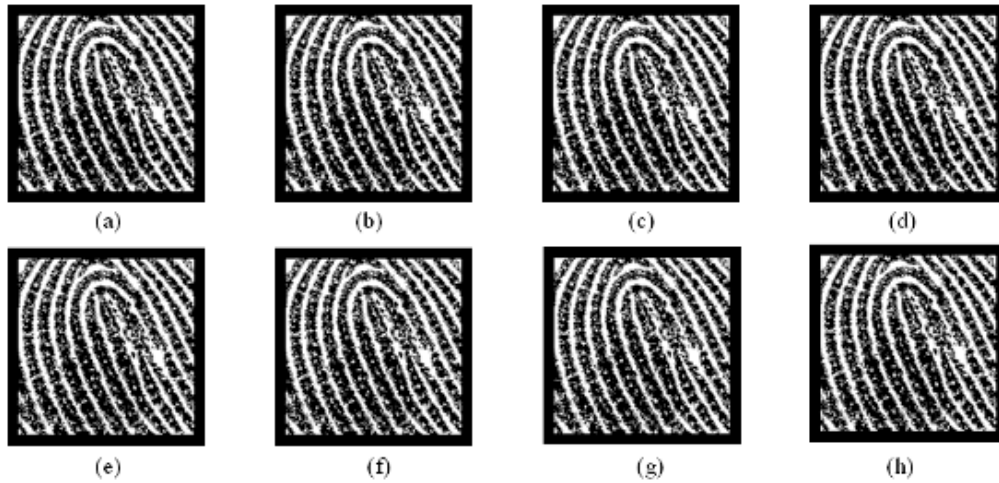


Fig. 5. Huellas digitales descryptadas a partir de huellas digitales encriptadas con diferentes niveles de cuantización: (a) $NC=8$, (b) $NC=9$, (c) $NC=10$, (d) $NC=11$, (e) $NC=12$, (f) $NC=13$, (g) $NC=14$, (h) $NC=15$.

TABLA 3. MSE Y SNR PARA DIFERENTES NIVELES DE CUANTIZACIÓN CUANDO LA HUELLA DIGITAL ENCRYPTADA ES GUARDADA Y POSTERIORMENTE DESENCRIPTADA.

NC	8	9	10	11	12	13	14	15
MSE	1.1218	0.3221	7.6341e-2	1.2056e-3	0	0	0	0
SNR	18638.732	64913.2424	2.7388e5	1.7342e7	∞	∞	∞	∞

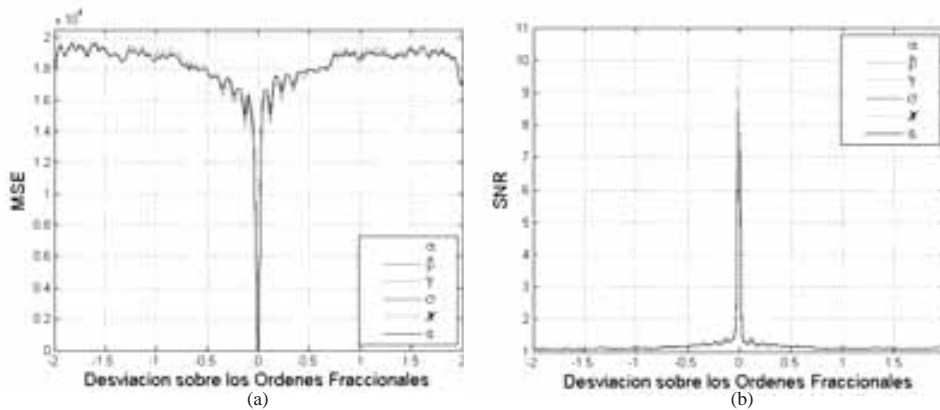


Fig. 6. Prueba de desviación de los órdenes fraccionales de las DFrFTs sobre los valores correctos del proceso de descryptación: (a) MSE, (b) SNR.

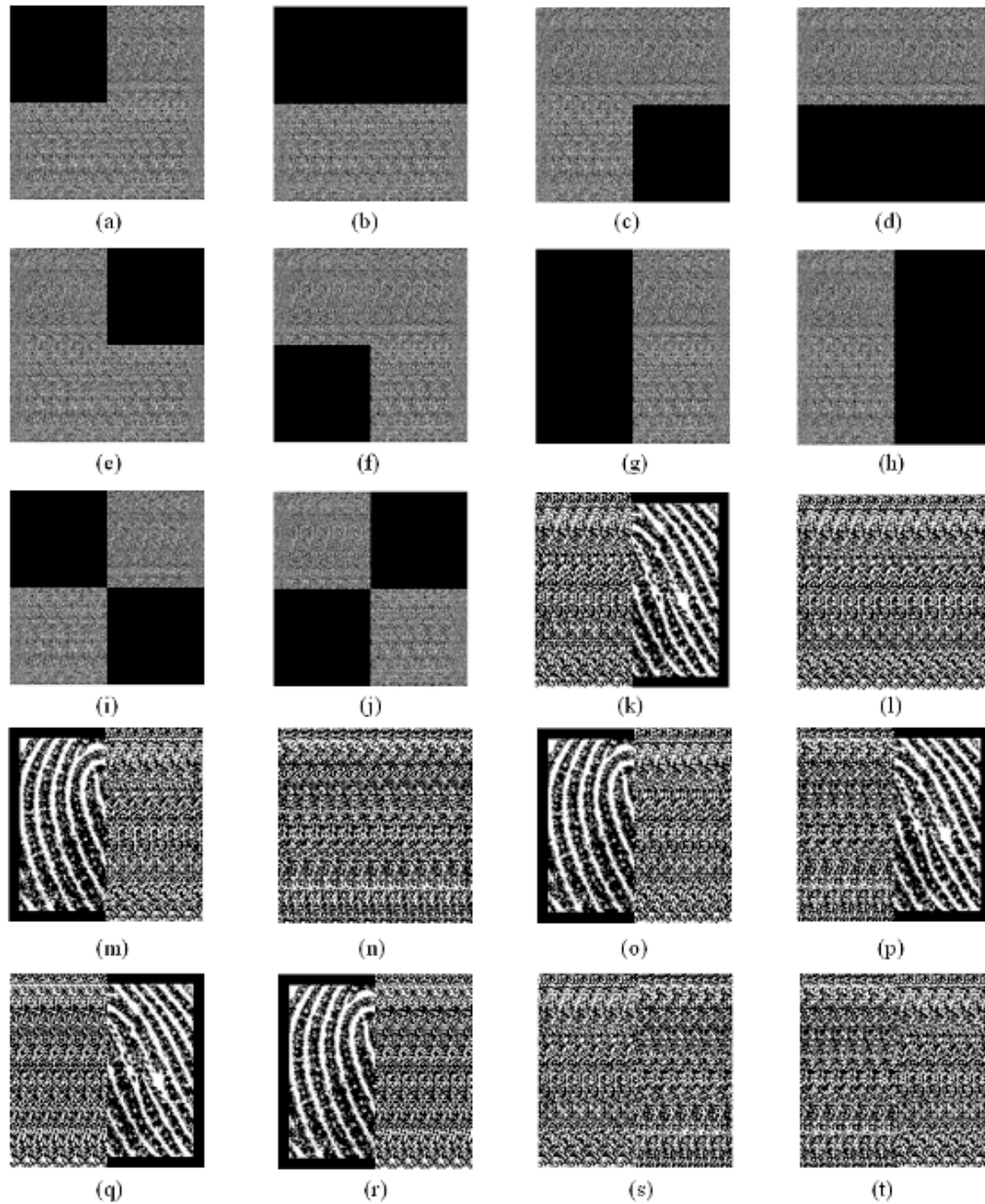


Fig. 7. Prueba de oclusión de píxeles sobre la huella digital encriptada, las primeras 10 huellas digitales corresponden a las diferentes oclusiones del 25% y 50% de píxeles (huellas encriptadas) y las restantes 10 huellas digitales son las imágenes desencriptadas para las diez oclusiones propuestas, respectivamente: (k) $MSE=1.0953e4$, $SNR=1.9089$, (l) $MSE=2.1938e4$, $SNR=0.9531$, (m) $MSE=1.0559e4$, $SNR=1.981$, (n) $MSE=2.1252e4$, $SNR=0.9838$, (o) $MSE=1.0985e4$, $SNR=1.9032$, (p) $MSE=1.0693e4$, $SNR=1.955$, (q) $MSE=1.2652e4$, $SNR=1.6525$, (r) $MSE=1.2457e4$, $SNR=1.6757$, (s) $MSE=2.1512e4$, $SNR=0.9719$, (t) $MSE=2.1678e4$, $SNR=0.9644$.

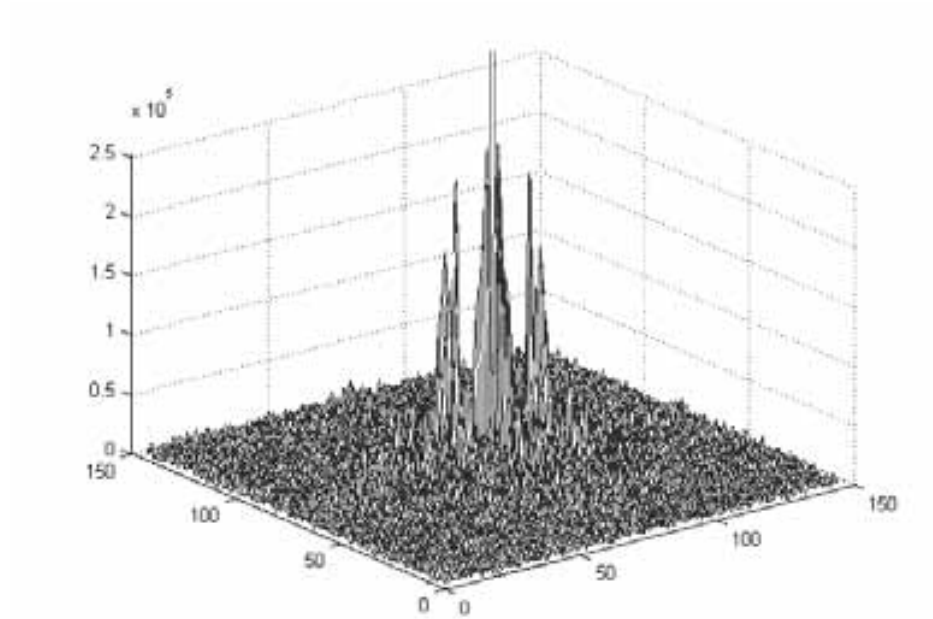


Fig. 8. Espectro de la huella digital de prueba.

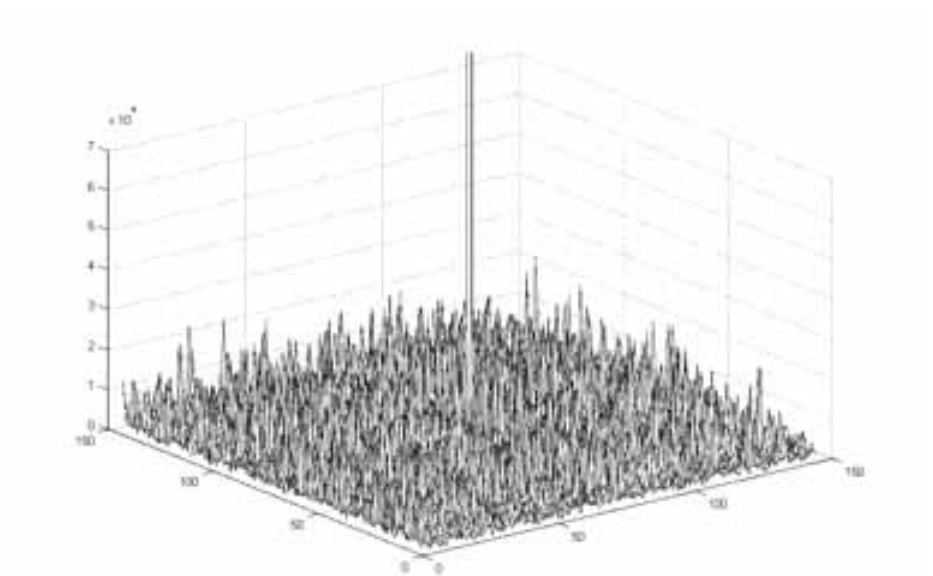


Fig. 9. Espectro de la huella digital encriptada

Modelado directo e inverso de imágenes infrarrojas de fase usando la teoría de cuádrupolos térmicos en termografía de fase pulsada

Hernán Benítez, Member, IEEE, Abdel Hakim Bendada, Member, IEEE, Clemente Ibarra-Castanedo, Member, IEEE, Xavier Maldague, Member, IEEE,

Resumen—La termografía de fase pulsada (TFP) es una técnica bien conocida en el ensayo termográfico no destructivo de materiales. El cálculo del contraste de fase permite la estimación de la profundidad de defectos en materiales encontrando la frecuencia límite que se define como la máxima frecuencia a la cual defecto puede ser detectado. Sin embargo, no existe una expresión analítica para las curvas de fase que permita estudiar la influencia del ruido y los parámetros de los defectos como la profundidad z_1 y la resistencia térmica R en el algoritmo de estimación de la frecuencia límite. En este trabajo se desarrolla un modelo simple analítico para las curvas de contraste térmico y se propone un algoritmo de estimación simple y completo sobre la expresión analítica que permite la estimación de la profundidad y la resistencia térmica.

Palabras Claves—Termografía de Fase pulsada, problemas inversos, cuádrupolos térmicos

I. INTRODUCCIÓN

LA Termografía de Fase Pulsada (TFP) [1] es una técnica de Evaluación No Destructiva (END) apropiada para la inspección rápida de grandes superficies y está basada en la Transformada Rápida de Fourier (FFT). Esta técnica brinda la posibilidad de extraer datos de amplitud y fase a partir de una secuencia de termogramas en bruto. Las imágenes de fase son de mucho interés ya que permiten una mejor detección de defectos y están menos afectadas por las variaciones de emisividad y calentamiento [2]. El cálculo de la profundidad de un defecto es posible a partir de las curvas de contraste de fase si se determina la frecuencia límite f_b que es la máxima frecuencia en la cual el defecto puede ser aún detectado [3], [4]. La ecuación 1 calcula la profundidad del defecto z_1 :

$$z_1 = C_1\mu + C_2 = C_1\sqrt{\frac{\alpha}{\pi f_b}} + C_2 \quad (1)$$

en donde μ es la longitud de difusión térmica y C_1 y C_2 son constantes de correlación estimadas a partir de datos experimentales y α es la difusividad térmica del material. Sin embargo, no existe una expresión analítica para las curvas de fase que permita estudiar la influencia del ruido generado por medición ni las variaciones de los parámetros de los defectos tales como profundidad z_1 o resistencia térmica

R . En este trabajo se desarrolla un modelo analítico simple para las curvas de contraste de fase. El modelo está basado en la teoría de cuádrupolos térmicos [5] cuyo formalismo está dado en el dominio de Laplace. Después de aplicar la transformada de Laplace, el problema puede ser resuelto en el dominio de Laplace para después ser transformado de vuelta al espacio original por transformada inversa [6]. Por esta razón, este método es usado para la solución del problema directo (cálculo de la respuesta térmica de un sistema) y del problema inverso (estimación de parámetros del sistema con base en la respuesta térmica) [7], [8], [9], [10], [11]. Sin embargo, dada la base analítica de este método se restringe a problemas lineales con geometrías simples.

En este estudio la solución en el dominio de Laplace se traslada al dominio de Fourier con el fin de obtener los datos de fase. En esta solución, los parámetros como z_1 y R aparecen explícitamente. Se estudian dos tipos de defectos: variaciones del grosor del material y delaminaciones. Finalmente, se propone un algoritmo de estimación, más simple y completo que el de la frecuencia límite, basado en la expresión analítica obtenida que permite la estimación de la profundidad z_1 y de la resistencia térmica R . Este trabajo contribuye al desarrollo tecnológico del país a través de la propuesta de una metodología de inspección termográfica de materiales que puede ser empleada en el diagnóstico de estructuras civiles como puentes y edificios.

II. EXPRESIÓN ANALÍTICA PARA EL CONTRASTE DE FASE Y ESTIMACIÓN DE LA PROFUNDIDAD DEL DEFECTO

Usando el modelo 1D del formalismo de cuádrupolos térmicos, se lleva a cabo un estudio de una muestra de dos capas hechas del mismo material y separadas por una resistencia de contacto R . Este espécimen es estimulado con un impulso de calor Delta. Si denominamos $\Theta(x, \omega)$ y $\Phi(x, \omega)$ las transformadas de Fourier de la temperatura y del flujo de calor entonces la matriz de transferencia que describe la transferencia de calor a través de la muestra completa está dada por:

$$\begin{bmatrix} \Theta(0, \omega) \\ \Phi(0, \omega) \end{bmatrix} = \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} 1 & R \\ 0 & 1 \end{bmatrix} \begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix} \begin{bmatrix} \Theta(z, \omega) \\ \Phi(z, \omega) \end{bmatrix} \quad (2)$$

en donde:

H. Benítez, Profesor Departamento Ciencias e Ingeniería de la Computación, Grupo DESTINO, Pontificia Universidad Javeriana-Cali
C. Ibarra-Castanedo, Investigador Posdoctoral, Universidad Laval, Québec, Canada

H. Bendada, X. Maldague, Profesores Universidad Laval, Québec, Canada

$$A_i = \left(\cosh \sqrt{\frac{j\omega}{\alpha}} z_i \right) = D_i \quad B_i = \left(\frac{\sinh \sqrt{\frac{j\omega}{\alpha}} z_i}{\lambda \sqrt{\frac{j\omega}{\alpha}}} \right) \quad (3)$$

$$C_i = \lambda \sqrt{\frac{j\omega}{\alpha}} \sinh \left(\sqrt{\frac{j\omega}{\alpha}} z_i \right)$$

y λ es la conductividad térmica del material. Es importante aclarar que la ecuación 2 se obtiene del formalismo de cuadrupolos original, el cual se lleva a cabo en el dominio de Laplace reemplazando la variable de Laplace p por $j\omega$ (j es la variable imaginaria) con el fin de describir la transferencia de calor en régimen modulado a una frecuencia moduladora ω . A partir de la ecuación 2 se puede derivar una expresión para la fase:

$$\arg \left(\frac{\Theta(0, \omega)}{\Phi(0, \omega)} \right)_{def} = \arg \left(\frac{A_1 A_2 + B_1 C_2 + A_1 R C_2}{C_1 A_2 + D_1 C_2 + C_1 R C_2} \right) \quad (4)$$

que corresponde al caso defectuoso. El caso no defectuoso resulta de hacer $R=0$ en la ecuación 4. Luego, la expresión analítica para el contraste de fase en el modelo de única delaminación está definido por:

$$\Delta\Phi = \arg \left(\frac{A_1 A_2 + B_1 C_2 + A_1 R C_2}{C_1 A_2 + D_1 C_2 + C_1 R C_2} \right) - \arg \left(\frac{A_1 A_2 + B_1 C_2}{C_1 A_2 + D_1 C_2} \right) \quad (5)$$

Esta expresión está en términos de z_1 y R lo que permite un análisis fácil de estos parámetros con respecto a las curvas de contraste de fase y al algoritmo de estimación basado en la frecuencia límite. La figura 1 presenta el diagrama de una única delaminación que es modelado como una resistencia de contacto térmica (RCT) y la figura 2 muestra la variación de grosor de material causado por la falta o exceso de material.

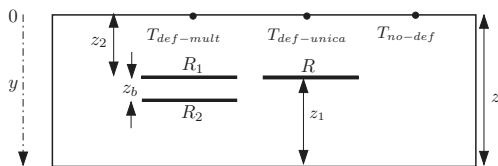


Fig. 1. Diagrama de las delaminaciones estudiadas: múltiple y única

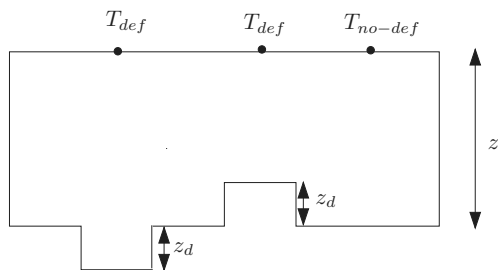


Fig. 2. Diagrama de las variaciones de espesor en el material inspeccionado

Estos tipos de defectos son comunes en materiales laminados como el Plástico Reforzado con Fibra de Carbono (PRFC) y las curvas de fase permiten distinguirlos. Las

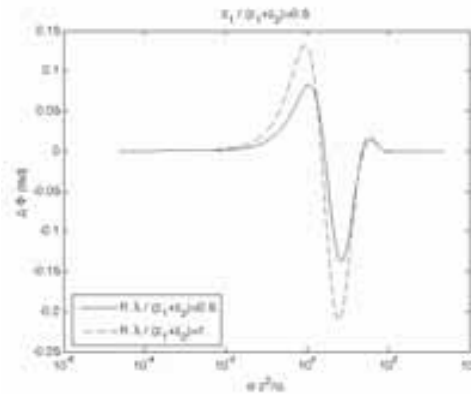


Fig. 3. Curvas de contraste de fase para delaminación única

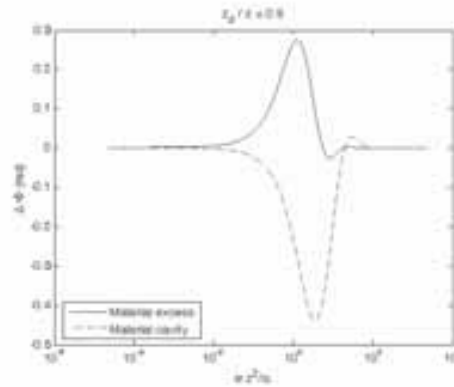


Fig. 4. Curvas de contraste de fase para variaciones de grosor

figuras 3 y 4 muestran las curvas de contraste de fase para una única delaminación y para variaciones en el grosor del material, respectivamente. Se observa que para el caso de las delaminaciones las curvas de contraste de fase tienen valores negativos después del segundo cruce por cero (frecuencia límite) y la curva toma de nuevo valores positivos a frecuencias más bajas antes de decrecer a cero. Por otro lado, la ausencia de material (corrosión) los valores del contraste de fase después del segundo cruce por cero son negativos, incrementándose a cero para frecuencias más bajas mientras que para el exceso de material estos son positivos decreciendo a cero para frecuencias más bajas. También se observa que contrario al caso de delaminación, no hay una tercera frecuencia de cruce por cero para el caso de variación de grosor del material (ausencia o exceso).

Con respecto a la estimación de la profundidad del defecto, se propone usar una optimización no lineal iterativa para encontrar estimativos de los parámetros (z_1 y R). Se usa la rutina **lsqnonlin** de la caja de herramientas Optimization Toolbox de Matlab [12] que es un método de región de subespacios y está basado en el método de Newton interior-reflectivo descrito en [13] y [14].

Las figuras 5 y 6 muestran los errores de estimación de profundidad en RCT y variación del grosor del material para la ecuación 1 del método tradicional y el algoritmo de estimación no lineal propuesto en el presente trabajo. Para este estudio comparativo los materiales para RCT y variación de grosor fueron PRFC y Plexiglas. Para el algoritmo de frecuencia límite C_1 y C_2 en la ecuación 1 son 1.73×10^{-5} y 6.56×10^{-5} para RCT y 2.01×10^{-4} y 2.7×10^{-4} para el caso de variación en el grosor del material [15]. El algoritmo de estimación no lineal entrega errores relativos menores que el algoritmo de frecuencia límite en todos los casos estudiados. Además, después de agregar ruido a las curvas de contraste de fase con distribución gaussiana especificada por $\mu = 0$ y $\sigma = 0.1$ los mayores errores relativos más altos obtenidos para RCT y variaciones del grosor del material fueron 3 % y 0.2 %, respectivamente.

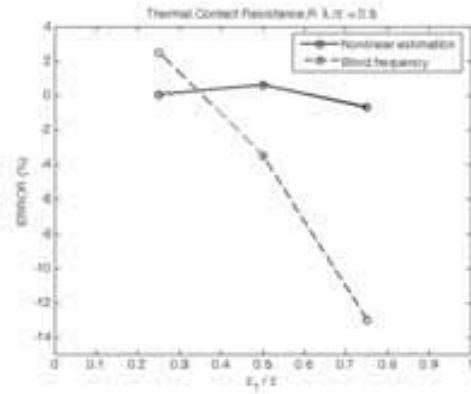


Fig. 5. Curvas de error relativo en RCT

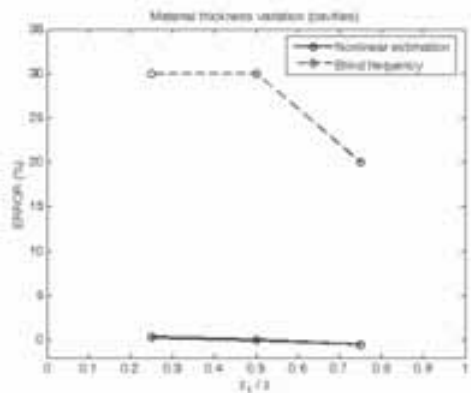


Fig. 6. Curvas de error relativo en variaciones de grosor de material

III. RESULTADOS EXPERIMENTALES

Con el fin de validar el algoritmo de estimación propuesto en la sección II se llevaron a cabo dos experimentos. El primero de estos en una muestra de Plexiglas con agujeros circulares superficiales para simular corrosión y el otro en una

muestra de PRFC con 2 mm de grosor e inserciones de Teflon para simular defectos. Las configuraciones de estas muestra se muestran en las figuras 7 y 8. La Tabla I ilustra los parámetros de muestreo y truncamiento empleados para la inspección de estas muestras donde N en el número de frames, $w(t)$ es el tiempo de adquisición y Δt es el intervalo de muestreo.

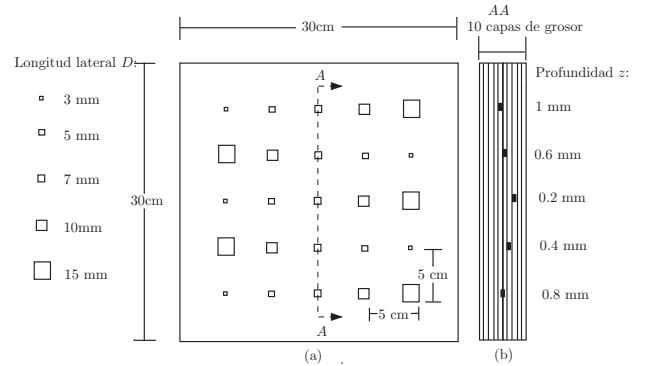


Fig. 7. Configuración de la muestra CFRP006

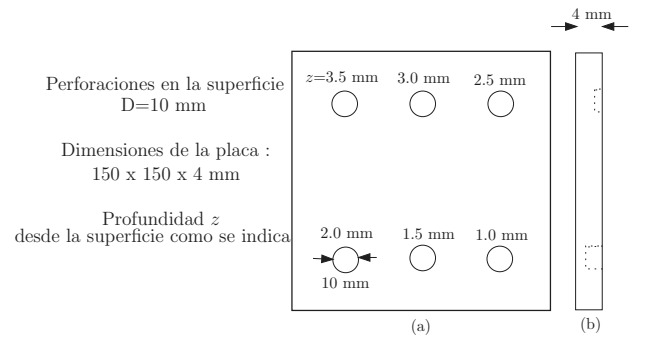


Fig. 8. Configuración de la muestra PLEXI014

TABLA I
PARÁMETROS DE TRUNCAMIENTO Y MUESTREO PARA LAS MUESTRAS CFRP006 Y PLEXI014

Parámetros de muestreo truncamiento	CFRP006	PLEXI014
N (frames)	989	600
$w(t)$ (s)	6.23	533
Δt (m.s)	6.3	880

A. PLEXI014

La figura 9 muestra las curvas de contraste de fase para todos los defectos en la muestra PLEXI014, estas curvas fueron obtenidas a partir de los puntos mostrados en el fasegrama (imagen de fase) de la figura 11 usando como área no defectuosa la región media derecha como se indica.

B. CFRP006

La figura 10 muestra las curvas de contraste de fase para defectos con $D_{eq} = 16.9$ mm en la muestra CFRP006,

estas curvas fueron extraídas de los puntos mostrados en el fasegrama de la figura 12 usando como área no defectuosa la región superior derecha de la muestra como se indica. El uso del diámetro equivalente definido como el diámetro de la circunferencia que tiene la misma área que la forma considerada, es propuesto para la comparación eventual con estudios similares de diferentes geometrías. Por ejemplo, el diámetro equivalente para un cuadrado con lado L es $D_{eq} = \frac{2L}{\sqrt{\pi}}$ [16].

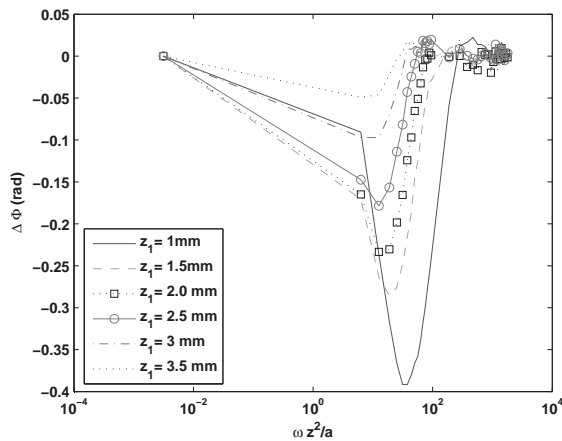


Fig. 9. Curvas de contraste de fase para muestra PLEXI014

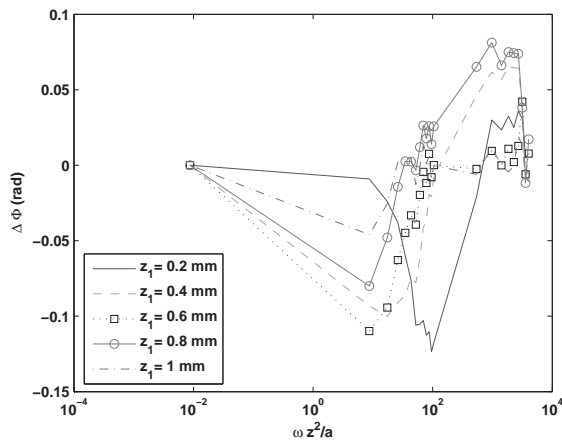


Fig. 10. Curvas de contraste de fase muestra CFRP006

Para la muestra de Plexiglas el valor inicial escogidos para el proceso de optimización es $z_1=1.3$ mm y para la muestra CFRP006 los valores iniciales son $z_1=0.5$ mm y $R=0.00062 \frac{m^2 K}{W}$. Durante la selección de estos valores iniciales se observó que los resultados son sensibles a los valores iniciales. Como se puede observar en la figura 13 para el caso de PLEXIGLAS las profundidades estimadas (z_{est}) están altamente correlacionadas ($R \approx 0.97$) con las profundidades reales z_{real} . Por otro lado, para la muestra CFRP006, el impacto de la profundidad del defecto se hace evidente en la figura 14 en donde los defectos más superficiales ($z_{real}=0.2$

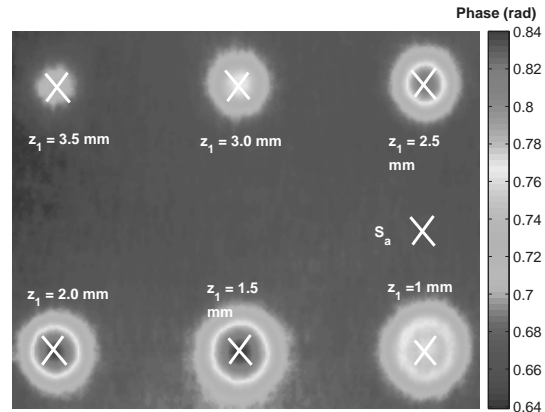


Fig. 11. Fasegrama (a $f = 0.0038$ Hz) de muestra PLEXI014

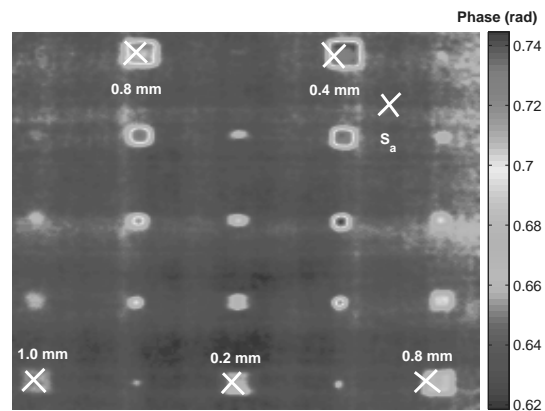
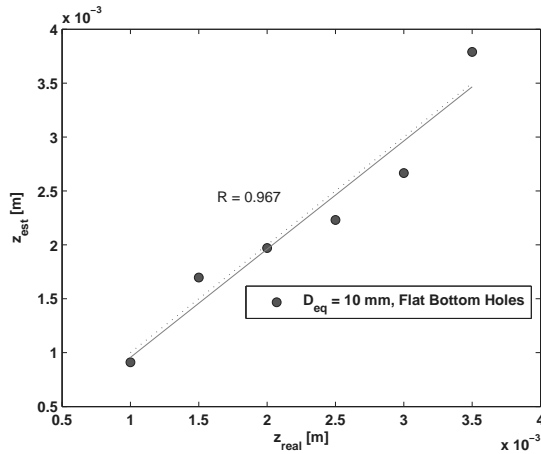
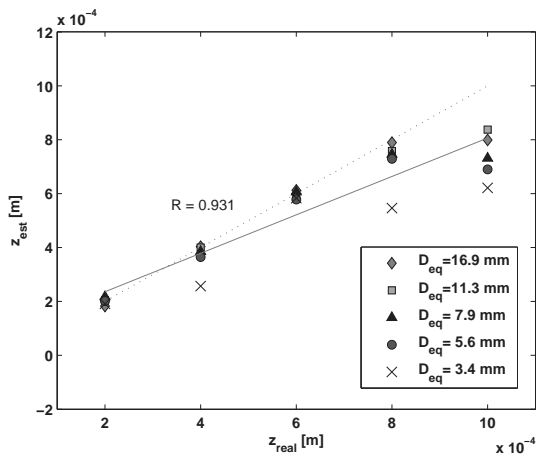
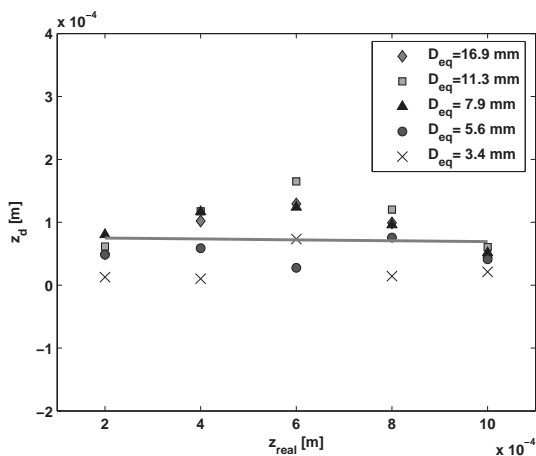


Fig. 12. Fasegrama (a $f = 0.32$ Hz) de muestra CFRP006

mm y 0.4 mm) forman un conglomerado compacto, mientras que para defectos más profundos estos conglomerados son dispersos. Además, la estimación de defectos para defectos pequeños ($D_{eq} = 3.4$ mm) a $z=0.4$ mm, 0.8 mm y 1 mm tiene errores grandes. La figura 15 muestra el grosor estimado de las inserciones de Teflon ($z_d = R_{est} \lambda_{Teflon}$) en la muestra CFRP006. Se puede observar que el grosor de las muestras de defectos pequeños ($D_{eq} = 3.4$ mm) son subestimados y que los datos están dispersos de la media indicando que la estimación de la resistencia térmica es afectada por la profundidad y el tamaño del defecto. Por otro lado, al comparar el método descrito en [3] con la metodología propuesta en el presente trabajo se observa que este último ofrece la posibilidad de estimar el grosor del defecto además de su profundidad y que está basado en un modelo teórico para el cual no es necesario calcular constantes.

IV. CONCLUSIONES

El formalismo de los cuadrupolos térmicos permite tener expresiones simples y explícitas para analizar los efectos de

Fig. 13. Resultados de la estimación de z_1 para PLEXI014Fig. 14. Resultados de la estimación de z_1 para CFRP006Fig. 15. Estimación del grosor z_d de inserciones en CFRP006

defectos en materiales que se manifiestan ya sea como RCT o variaciones en el grosor del material pueden ser discriminados basados en la forma de sus curvas de contraste térmico. Se presentó y validó un algoritmo de estimación basado en mínimos cuadrados no lineales con datos experimentales de muestras de Plexiglas y CFRP para tipos de defectos como variaciones en el grosor del material y delaminaciones (RCT). Estos resultados mostraron que el algoritmo de estimación puede estimar la profundidad de defectos en el caso de variaciones de grosor con una exactitud razonable y que para un material anisotrópico como el PRFC con inserciones de Teflon la relación $\frac{D_{eq}}{z_1}$ de los defectos afecta la estimación de la profundidad y de la resistencia térmica.

BIBLIOGRAFÍA

- [1] X. Maldague and S. Marinetti, "Pulse phase infrared thermography," *Journal of Applied Physics*, vol. 79, no. 5, pp. 2694–2698, 1996.
- [2] G. Busse and A. Rosenwaig, "Subsurface imaging with phase sensitive modulated thermography," *Appl. Phys. Lett.*, vol. 36, pp. 815–816, 1980.
- [3] C. Ibarra-Castanedo and X. Maldague, "Interactive methodology for optimized defect characterization by quantitative pulsed phase thermography," *Res. Nondestr. Eval.*, vol. 16, no. 4, pp. 1–19, 2005.
- [4] Y. hong Li, Y. jin Zhao, L. chun Feng, and C. lin Zhang, "Study on pulsed phase analysis of depth measurement for infrared thermal wave nondestructive evaluation," in *Infrared Materials, Devices and Applications*, Y. C. H. G. Jean-Pierre, Ed., vol. 6835, 2008.
- [5] D. Maillet, S. André, J. Batsale, A. Degiovanni, and C. Moyne, *Thermal quadrupoles: solving the heat equation through integral transforms*, J. Wiley and Sons, Eds. West Sussex, England: John Wiley and Sons, 2000.
- [6] H. Stehfest, "Remarks on algorithm 368: Numerical inversion of laplace transforms," *Communications of the A.C.M.*, vol. 13, no. 1, pp. 47–49, 1970.
- [7] A. Bendada, D. Maillet, J. Batsale, and A. Degiovanni, "Reconstitution of a non uniform interface thermal resistance by inverse heat conduction application to infrared thermography testing," *Inverse Problems in Engineering*, vol. 6, pp. 79–123, 1998.
- [8] J. Beck and K. Woodbury, "Inverse problems and parameter estimation: integration of measurements and analysis measurement," *Measurement. Science. Technology*, vol. 9, pp. 839–847, 1998.
- [9] D. Maillet, A. Houlbert, S. Didierjean, A. Lamine, and A. Degiovanni, "Non-destructive evaluation of delaminations in a laminate: part I, identification by measurement of thermal contrast," *Inverse problems and parameter estimation: integration of measurements and analysis Measurement*, vol. 9, pp. 137–153, 1993.
- [10] V. Vavilov, D. Kourtenkov, E. Grinzato, S. Marinetti, and C. Rissan, "Inversion of experimental data and thermal tomography using thermo.heat and termidge software," in *Quantitative Infrared Thermography*, 1994.
- [11] C. Ibarra-Castanedo, D. González, F. Galmiche, X. Maldague, and A. Bendada, "Discrete signal transforms as a tool for processing and analyzing pulsed thermographic data," in *Proceedings SPIE Thermosense XXVIII*, J. Miles, G. Raymond Peacock, Kathryn M. Knettel Editors, 2006.
- [12] *Mathworks, Inc. Optimization toolbox, V 3.1, (2006)*.
- [13] T. Coleman and Y. Li, "An interior trust region approach for nonlinear minimization subject to bounds," *SIAM Journal on Optimization*, vol. 6, pp. 418–445, 1996.
- [14] —, "On the convergence of reflective newton methods for large-scale nonlinear minimization subject to bounds," *Mathematical Programming*, vol. 67, no. 2, pp. 189–224, 1994.
- [15] C. Ibarra-Castanedo, "Quantitative subsurface defect evaluation by pulsed phase thermography: Depth retrieval with the phase," Ph.D. dissertation, Université Laval, Electrical Engineering and Computing Department, 2005.
- [16] C. Ibarra-Castanedo, N. Avdelis, E. Grinzato, P. Bison, S. Marinetti, L. Chen, M. Genest, and X. Maldague, "Quantitative inspection of non-planar composite specimens by pulsed phase thermography," *QIRT*, vol. 3, no. 1, pp. 25–40, 2006.

los parámetros de los defectos (z_1 y R) sobre las curvas de contraste de fase para diferentes tipos de defectos. Además, los

Automatic analysis for the chemical testing of urine examination using digital image processing techniques

Juan M. Vilardy, Jose C. Peña, Miller F. Daza, Cesar O. Torres, Lorenzo Mattos.

Abstract— For to make the chemical testing of urine examination a dipstick is used, which contains pads that have incorporated within them the reagents for chemical reactions for the detection of a number from substances in the urine. Urine is added to the pads for reaction by dipping the dipstick into the urine and then slowly withdrawing it. The subsequent colorimetric reactions are timed to an endpoint; the extent of colors formation is directly related to the level of the urine constituent. The colors can be read manually by comparison with color charts or with the use of automated reflectance meters. The aim of the System described in this paper is to analyze and to determine automatically the changes of the colors in the dipstick when this is retired of the urine sample and to compare the results with color charts for the diagnosis of many common diseases such as diabetes. The system consists of: (a) a USB camera. (b) Computer. (c) Software Matlab v7.4. Image analysis begins with a digital capturing of the image as data. Once the image is acquired in digital format, the data can be manipulated through digital image processing. Our objective was to develop a computerised image processing system and an interactive software package for the backing of clinicians, medical research and medical students.

Keywords— Color Image Processing, Computer Vision, Digital Image Processing, Testing of Urine.

I. INTRODUCTION

One of the most prominent applications for the computer vision is medical computer vision or medical image processing. This area is characterized by the extraction of information from image data for the purpose of making a

Este trabajo fue desarrollado en el Laboratorio de Óptica e Informática de La Universidad Popular Del Cesar (Valledupar, Colombia).

Juan M. Vilardy. Ingeniero Electrónico, Joven Innovador de COLCIENCIAS adjunto al Grupo de Óptica e Informática. (e-mail: vilardy.juan@unicesar.edu.co).

Jose C. Peña. Ingeniero de Sistemas, Estudiante de Doctorado de Ingeniería, Universidad de Antioquia. (e-mail: mattos.jc@gmail.com).

Miller F. Daza. Estudiante de Ingeniería Electrónica, Universidad Popular del Cesar. (e-mail: millerdaza1@hotmail.com).

Cesar O. Torres. Dr. en Física. Docente asociado a La Universidad Popular Del Cesar, Director del Grupo de Óptica e Informática. (e-mail: torres.cesar@caramail.com).

Lorenzo. Mattos. Msc. en Informática. Docente asociado a La Universidad Popular Del Cesar, Coordinador del Grupo de Óptica e Informática. (e-mail: mattos.lorenzo@caramail.com).

medical diagnosis of a patient. Generally, image data is in the form of microscopy images, X-ray images, angiography images, ultrasonic images, and tomography images. An example of information which can be extracted from such image data is detection of tumours, arteriosclerosis or other malign changes. It can also be measurements of organ dimensions, blood flow, etc [1]. In this paper, the vision computer it applied to the urinalysis, for the extraction of information from image data of the dipstick.

A urinalysis is a group of manual and/or automated qualitative and semi-quantitative tests performed on a urine sample. A routine urinalysis usually includes the following tests: color, transparency, specific gravity, pH, protein, glucose, ketones, blood, bilirubin, Nitrites, urobilinogen, and leukocyte esterase. Some laboratories include a microscopic examination of urinary sediment with all routine urinalysis tests. If not, it is customary to perform the microscopic exam, if transparency, glucose, protein, blood, Nitrites, or leukocyte esterase is abnormal. Routine urinalyses are performed for several reasons: general health screening to detect renal and metabolic diseases, diagnosis of diseases or disorders of the kidneys or urinary tract, monitoring of patients with diabetes. In addition, quantitative urinalysis tests may be performed to help diagnose many specific disorders, such as endocrine diseases, bladder cancer, osteoporosis, and porphyrias (a group of disorders caused by chemical imbalance) [2].

II. THE CHEMICAL TESTING OF URINE EXAMINATION

Urinalysis involves both physical and chemical evaluation. Physical examination includes inspection of the color, turbidity, specific gravity and osmolality, and pH. Normal urine is pale yellow in color, although it may vary considerably due to food intake, hydration status, metabolites, and infection. Fresh urine is usually clear, although it may be cloudy after meals or with milk intake due to precipitation of phosphates in the urine. Pyuria, excessive white blood cells in the urine due to infection, will cause turbidity but also causes a foul odor in the urine. Microscopic analysis will differentiate between phosphate crystals and leukocytes. Other causes of urine turbidity are chyluria (lymph fluid mixed with urine), lipiduria, hyperoxaluria, and hyperuricosuria.

Many chemical tests can be performed on a small quantity

of urine by using a dipstick. A dipstick is a piece of plastic to which pads of certain chemical reagents have been attached. Each pad will test for a particular substance in the urine. When the urine comes in contact with the reagents, a chemical reaction occurs which changes the color of the pad based upon how much of the substance is in the urine. The color of the pad is compared to a color chart and an approximate amount of the substance can be determined. There are several brands and types of dipsticks. Some may measure only one or two substances, while others may measure 8 or more [2].

Additional tests are available for measuring the levels of bilirubin, protein, glucose, ketones, and urobilinogen in urine. In general, these individual tests provide greater sensitivity; they therefore permit detection of a lower concentration of the respective substance. A brief description of the most commonly used dry dipstick tests follows [2]:

Leukocytes: The presence of white blood cells in the urine usually signifies a urinary tract infection, such as cystitis, or renal disease, such as pyelonephritis or glomerulonephritis.

Nitrites: Some disease bacteria, including the lactose-positive *Enterobacteriaceae*, *Staphylococcus*, *Proteus*, *Salmonella*, and *Pseudomonas* are able to reduce nitrate in urine to Nitrites. A positive test for Nitrites indicates bacteruria, or the presence of bacteria in the urine.

Urobilinogen: Urobilinogen is a substance formed in the gastrointestinal tract by the bacterial reduction of conjugated bilirubin. Increased urinary urobilinogen occurs in prehepatic jaundice (hemolytic anemia), hepatitis, and other forms of hepatic necrosis that impair the circulation of blood in the liver and surrounding organs. The urobilinogen test is helpful in differentiating these conditions from obstructive jaundice, which results in decreased production of urobilinogen.

Protein: Based upon a phenomenon called the "protein error of indicators," this test uses a pH indicator, such as tetrabromophenol blue, that changes color (at constant pH) when albumin is present in the urine. Albumin is important in determining the presence of glomerular damage. The glomerulus is the network of capillaries in the kidneys that filters low molecular weight solutes such as urea, glucose, and salts, but normally prevents passage of protein or cells from blood into filtrate. Albuminuria occurs when the glomerular membrane is damaged, a condition called **glomerulonephritis**.

pH: A combination of pH indicators (methyl red and bromthymol blue) react with hydrogen ions (H^+) to produce a color change over a pH range of 5.0 to 8.5. pH measurements are useful in determining metabolic or respiratory disturbances in acid-base balance. For example, kidney disease often results in retention of H^+ (reduced acid excretion). pH varies with a person's diet, tending to be acidic in people who eat meat but more alkaline in vegetarians. pH testing is also useful for the classification of urine crystals.

Blood: Red cells and hemoglobin may enter the urine from the kidney or lower urinary tract. Testing for blood in the urine detects abnormal levels of either red cells or hemoglobin, which may be caused by excessive red cell

destruction, glomerular disease, kidney or urinary tract infection, malignancy, or urinary tract injury.

Specific gravity: Specific gravity is a measure of the ability of the kidneys to concentrate urine by conserving water.

Ketones: Ketones are compounds resulting from the breakdown of fatty acids in the body. These ketones are produced in excess in disorders of carbohydrate metabolism, especially Type 1 diabetes mellitus. In diabetes, excess ketoacids in the blood may cause life-threatening acidosis and coma. These ketoacids and their salts spill into the urine, causing ketonuria. Ketones are also found in the urine in several other conditions, including fever; pregnancy; glycogen storage diseases; and weight loss produced by a carbohydrate-restricted diet.

Bilirubin: Bilirubin is a breakdown product of hemoglobin. Most of the bilirubin produced in humans is conjugated by the liver and excreted into the bile, but a very small amount of conjugated bilirubin is reabsorbed and reaches the general circulation to be excreted in the urine. The normal level of urinary bilirubin is below the detection limit of the test. Bilirubin in the urine is derived from the liver, and a positive test indicates hepatic disease or hepatobiliary obstruction.

Glucose (sugar): The glucose test is used to monitor persons with diabetes. When blood glucose levels rise above 160 mg/dL, the glucose will be detected in urine. Consequently, glycosuria (glucose in the urine) may be the first indicator that diabetes or another hyperglycemic condition is present. The glucose test may be used to screen newborns for galactosuria and other disorders of carbohydrate metabolism that cause urinary excretion of a sugar other than glucose.

III. THE EXPERIMENT SETUP

The experiment setup to make the chemical testing of urine examination was the following one:

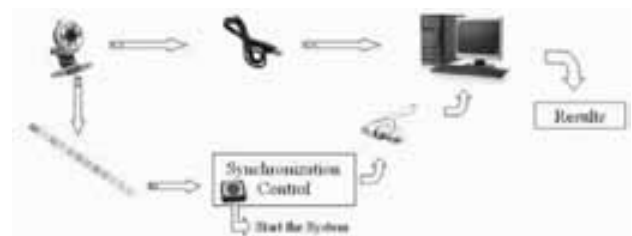


Fig. 1. The experiment setup for to make the chemical testing of urine examination.

The setup implemented has a synchronization module between the mechanical part where taking the images of the dipstick and the P.C that controls all the images acquisition system composed by a camera USB and a lamp of white light for the illumination of the dipstick. The taking of the images is controlled by the timing made by the P.C [3], so that the tests that can be made with dipstick take a time from different chemical reaction, therefore the images acquisition is due to fit in the times adapted for each test to make, altogether is necessary to take five images in the following times: 30, 40,

45, 60, and 120 seconds. Soon these images are processed to identify the resulting colors of the chemical reaction and finally to assign to the corresponding values with the identification of color for each one of the substances in the urine or made tests.

The dipstick to analyze is manufactured for Bayer. This dipstick has ten square pads used for the following tests: (a) Leukocytes, (b) Nitrites, (c) Urobilinogen, (d) Protein, (e) pH, (f) Blood, (g) Specific gravity, (h) Ketones, (i) Bilirubin, (j) Glucose (to sugar). An image of the dipstick used for to make the chemical testing of urine examination, was taken with the system implemented in figure 1:

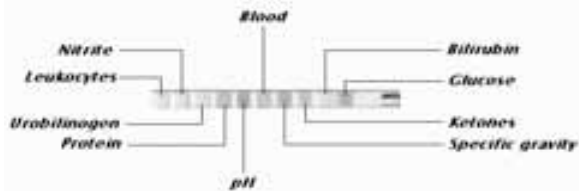


Fig. 2. The Bayer dipstick.

IV. EXPERIMENT RESULTS

When the button of the synchronization control is pressed, the program codified in the P.C on the programming platform Matlab v7.4 [3, 4], begins the capture of the five necessary images of the dipstick (these five images are necessary to make the chemical testing of urine according to the manufacturer of dipstick), immediately after it urine it makes contact with the dipstick. For a chemical testing of urine made with experimental setup of the figure 1, the five images of the dipstick, are:



Fig. 3. After 30 seconds, the image taken from the dipstick to analyze: Glucose and Bilirubin.

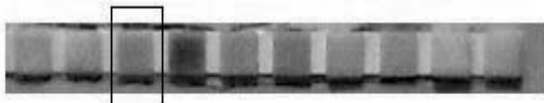


Fig. 4. After 40 seconds, the image taken from the dipstick to analyze: Ketones.

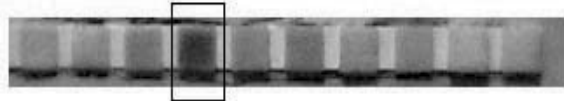


Fig. 5. After 45 seconds, the image taken from the dipstick to analyze: Specific gravity.

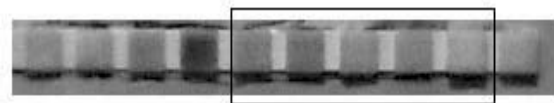


Fig. 6. After 60 seconds, the image taken from the dipstick to analyze: Blood, pH, Protein, Urobilinogen and Nitrites.



Fig. 7. After 120 seconds, the image taken from the dipstick to analyze: Leukocytes.

In order to obtain the colors identification it is necessary to locate the positions of the square pads, being applied borders detection with a Gaussian Laplacian filter, with the following convolution kernel or the mask [4]:

$$H = \begin{bmatrix} 0.0448 & 0.0468 & 0.0564 & 0.0468 & 0.0448 \\ 0.0468 & 0.3167 & 0.7146 & 0.3167 & 0.0468 \\ 0.0564 & 0.7146 & -4.9048 & 0.7146 & 0.0564 \\ 0.0468 & 0.3167 & 0.7146 & 0.3167 & 0.0468 \\ 0.0448 & 0.0468 & 0.0564 & 0.0468 & 0.0448 \end{bmatrix} \quad (1)$$

The filtering operation is done using the convolution in the temporal domain, Spatial filtering is a “local” operation in the sense that it modifies the value of each pixel according to the values of nearby pixels and the convolution kernel used (generally, this kernel is a square, uneven-sized matrix mask), using the Discrete Convolution operation, thus [5]:

$$I_{m,n} = \sum_{i=-r}^r \sum_{j=-r}^r H_{i,j} G_{m-i,n-j} = \sum_{i=-r}^r \sum_{j=-r}^r H_{-i,-j} G_{m+i,n+j} \quad (2)$$

From the previous equation we have: $G_{m,n}$ as the image to be filtered of m rows and n columns, $H_{i,j}$ is the convolution kernel or the mask, which is assumed of uneven size (for the purpose of placing the pixel to be changed in the center of the mask, and conducting the filtering process as previously mentioned) with $(2r+1)(2r+1)$ coefficients, that is $2r+1$ rows and $2r+1$ columns, and $I_{m,n}$ is the result of the filtering operation [5].

The borders detection with the mask previously mentioned, is:



Fig. 8. borders detection in a dipstick.

After finding the ten positions of the square pads of the dipstick, it is come to make the color comparison by each one of the channels: Red (R), Green (G) and Blue (B), being used the average pixel in each channel for each of the square pads that represent the different tests of the urine (Leukocytes, Nitrites, Urobilinogen, Protein, pH, Blood, and others). The result for the chemical testing of urine using the images of the figures 3 to the figure 7, and applying the automatic colors comparison by channels RGB, it is:

Leukocytes: Negative
 Nitrites: Negative
 Urobilinogen: 2
 Protein: 30 + mg/dl
 pH: 6.5
 Blood: Negative
 Specific gravity: 1.010
 Bilirubin: Moderate ++
 Ketones: Negative
 Glucose: Negative

The later analysis of the results must be made by a specialist in the urinalysis area. The color letter that was used for the colors comparison it show in the figure 9.

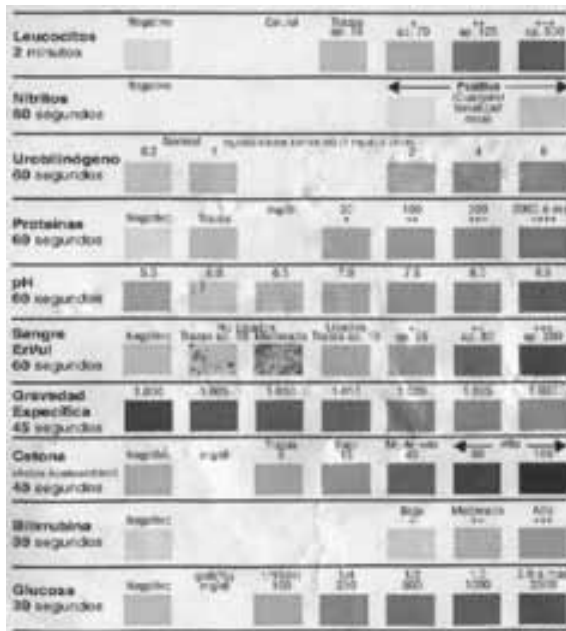


Fig. 9. Color letter used in the colors comparison.

V. CONCLUSIONS

The use of the computer vision for to make the chemical testing of urine examination, improves the precision of the results of this examination that are made by the vision of the human, eliminating all a series of factors of the surroundings that can influence negatively in the quality of the results of the examination: different conditions of lighting system in the work site, capacity of discernment of colors individually different from the people who make the evaluation, diminution of the capacity to concentrate in case of great series of samples and differences in the exactitude of observation of the time of reaction of the dipstick. It is possible to extend the number of dipsticks to analyze so that the images that are taken to identify them colors in the dipsticks are not very great and the amount of pixels to analyze is small, thus managing to analyze more dipsticks in the same time of analysis of a dipstick. Finally this algorithm-software implemented can be used for clinical investigations and professional of the urinalysis area.

ACKNOWLEDGMENTS

The work was supported by the Universidad Popular del Cesar.

REFERENCES

- [1] G. Pajares, J. De la Cruz, *Visión por Computadora*, Ed. Alfaomega, pp 65 – 179. Colombia, noviembre, 2004.
- [2] Strasinger SK. *Líquidos corporales y análisis de orina*. México: Editorial El Manual Moderno, 1991: 1-195.
- [3] The MathWorks, *MATLAB Image Acquisition Toolbox help*. http://www.mathworks.com/access/helpdesk/help/pdf_doc/imaq/imaq_p rint.pdf. March 2007.
- [4] The MathWorks, *MATLAB Image Processing Toolbox help*. http://www.mathworks.com/access/helpdesk/help/pdf_doc/imaq/imaq_p rint.pdf. March 2007.
- [5] Jahne, *Digital Image Processing*. Ed. Springer-Verlag. Alemania, 2003.

Optimización de la curva de distorsión utilizando cuantización vectorial de estado finito

C. H. Valencia, R. E. Ramírez, M. S. Dutra

Resumen—Dentro de las consideraciones especiales para el tratamiento y transmisión de información de un punto a otro, diferentes técnicas son utilizadas para obtener facilidades representativas en ahorro de tiempo, energía, procesamiento[1] y a su vez brindar seguridad. En este artículo es presentada la cuantización vectorial de estado finito como una técnica alternativa para conseguir altas tasas de compresión de datos y niveles de codificación que brinden seguridad al usuario. La característica más importante es la reducción de la curva de distorsión presentada en la cuantización vectorial tradicional, permitiendo obtener codificaciones de mejor calidad utilizando un número menor de bits y un número de iteraciones notablemente más bajo que con los esquemas tradicionales.

Palabras Claves— Curva de Distorsión, Reducción de Número de Bits y Cuantización Vectorial de Estado Finito.

I. INTRODUCCIÓN

LOS métodos de interés a ser tratados en este artículo para tareas de codificación y compresión de datos presentan numerosas ventajas frente a modelos clásicos que buscan desarrollar las mismas tareas. Es por eso que sobre el modelo de cuantización vectorial se han desarrollado evoluciones. Como objetivo este artículo presenta la optimización de la curva de distorsión disminuyendo tanto el número de bits requeridos así como el costo computacional.

Durante el desarrollo del siguiente artículo serán presentados en la parte II Conceptos generales de Cuantización Vectorial y en la parte III los resultados obtenidos para la Optimización de la Curva de Distorsión.

II. CUANTIZACIÓN VECTORIAL DE ESTADO FINITO

César Hernando Valencia Niño es Docente de la División de Ingenierías en la Universidad Santo Tomás - Seccional Bucaramanga - Colombia, Candidato a Magíster en Ciencias en Ingeniería Mecánica por la Universidad Federal de Rio de Janeiro - Brasil, Investigador de la Unidad de Investigación de Telecomunicaciones (UNITEL/USTA) y el Laboratorio de Diseño de Maquinas y Robótica (LabRob/COPPE/UFRJ); correo electrónico: cesar.valencia@ufrj.br

Ricardo Ramírez Heredia es Docente del Departamento de Ingeniería Mecánica y Mecatrónica en la Universidad Nacional de Colombia, Candidato a Doctor en Ciencias en Ingeniería Mecánica por la Universidad Federal de Rio de Janeiro - Brasil, Investigador del Grupo de Robótica (UNAL) y el Laboratorio de Diseño de Maquinas y Robótica (LabRob/COPPE/UFRJ); correo electrónico: rramirez@ufrj.br, reramirez@unal.edu.co

Max Suell Dutra es Docente del Programa de Ingeniería Mecánica en la Universidad Federal de Rio de Janeiro - Brasil y Director del Laboratorio de Diseño de Maquinas y Robótica (LabRob/COPPE/UFRJ); correo electrónico: max@mecanica.coppe.ufrj.br

A. Cuantización Vectorial Recursiva

Son conocidos como cuantizadores vectoriales recursivos los cuantizadores con retroalimentación, conocidos también como sistemas con memoria.

En la Figura 1, son presentados los diagramas de bloques para codificadores y decodificadores recursivos[2]. En ellos los índices de símbolos generados por canal para el codificador dependen tanto de los vectores de entrada como de los vectores *a priori*, correspondientemente a esto la reproducción generada por el codificador es dependiente del símbolo actual y de los símbolos anteriores.

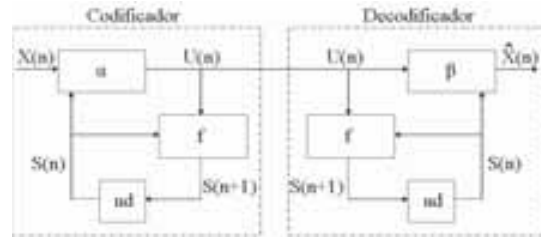


Fig. 1. Codificadores y decodificadores recursivos.

La salida de la función $S(n+1)$ está determinada por el canal de símbolos $U(n)$ y la secuencia de estados $S(n)$ para un mapeo f de tal forma como es presentada en (1).

$$S(n+1) = f(U(n), S(n)), n = 0, 1, \dots, \quad (1)$$

De tal forma el mapeo realizado por el codificador α entregará como resultado (2), que esta compuesta por los vectores de entrada $X(n)$ y el resultado de la secuencia de estado $S(n)$.

$$U(n) = \alpha(X(n), S(n)) \quad (2)$$

El decodificador realiza un mapeo β usando la secuencia de estados actuales $S(n)$ y la salida del codificador $U(n)$ para generar la reproducción[3] tal como es presentado en (3).

$$\hat{X}(n) = \beta(U(n), S(n)) \quad (3)$$

Es importante aclarar que para el estado inicial $S(0)$ la salida del codificador va ser exactamente el vector de entrada. El codificador trabaja con la condición de vecino más próximo o distorsión mínima según (4), la colección de todas las reproducciones posibles para todos los canales son obtenidas por la función del decodificador para un estado fijo S .

$$\alpha(X, S) = \min_U^{-1} d(X, \beta(U, S)) \quad (4)$$

$$\Delta = \frac{1}{L} \sum_{i=1}^L d(X(i), \hat{X}(i)) \quad (5)$$

La Ecuación (5) es utilizada para determinar la distorsión media entre los vectores de entrada y las reproducciones generadas por el decodificador.

B. Procesos de Estado Finito

Los procesos de estado finito[4] utilizados para cuantizadores vectoriales son una aplicación de sistemas recursivos, con un número finito de estados. Los estados $S=\{\sigma_1, \dots, \sigma_K\}$ para cada uno de los estados $S(n)$ pueden tomar valores en S . Los estados del codificador FSVQ pueden ser considerados como las formas de predicción del nuevo vector de entrada, basándose en los vectores codificados en el pasado o en todas las reproducciones posibles del decodificador. Cada elemento determina el diccionario[5] y selecciona el mejor diccionario que es equivalente a la predicción del comportamiento del próximo vector de entrada. El estado del diccionario es la colección de todas las palabras según (6).

$$\hat{X} + \hat{e} \tag{6}$$

Donde \hat{X} es la predicción actual para el vector actual y \hat{e} son las palabras en el diccionario residual. El estado actual es utilizado para especificar cual diccionario es usado y el índice en el canal para ser combinados con el estado actual y así determinar el próximo estado y próximo diccionario. También es conocido como (*Switched Vector Quantizer – SVQ*) o cuantización vectorial conmutada[6], en la Fig. 2, están los esquemas para cuantización vectorial de estado finito.

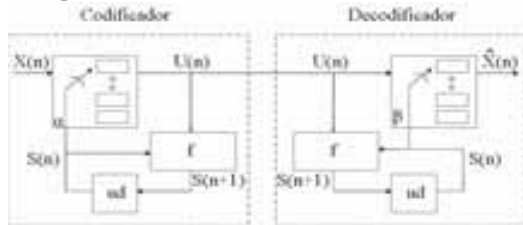


Fig. 2. Codificador y decodificador de estado finito.

$$U(n) = \eta_{S_n}(X(n)) \tag{7}$$

$$\hat{X} = \zeta_{S_n}^{-1}(U(n)) \tag{8}$$

En las Ecuaciones (7) y (8) son presentadas respectivamente las salidas del codificador y el decodificador, donde η_{S_n} representa la codificación de $X(n)$ con el diccionario $C(s)$ correspondiente al estado S y $\zeta_{S_n}^{-1}$ representa la reproducción decodificada para el índice i y el estado S .

TABLA I
ELEMENTOS DE UN CUANTIZADOR VECTORIAL

Elemento	Determinado por
Espacio de estados	$S=\{1,2,\dots,K\}$
Estado Inicial	$S(0)$
Función del decodificador	$\beta(U(n),S(n))$
Función estado siguiente	$f(U(n),S(n))$
Diccionarios de estado	$C(s)=\{\beta(U(n),S(n))\}$
Súper-Diccionarios	$C = \cup_{s \in S} C(s) = \{\beta(U(n),S(n)); U(n) \in N, S(n) \in S\}$

C. Tipos de Transiciones

En la cuantización vectorial de estado finito existen dos puntos de vista para la relación que existe entre la secuencia de estados y la secuencia de reproducción dentro de cualquier algoritmo[7]. Una de ellas es la que asocia una etiqueta a cada estado, así el decodificador realiza el mapeo γ produciendo una salida que depende del estado actual y del símbolo del canal solamente a través del estado siguiente, según (9).

$$\beta(U(n),S(n)) = \gamma(f(U(n),S(n))) \tag{9}$$

De esta forma la distorsión mínima debe satisfacer “(10)”.

$$d(X, \gamma(f(U(n),S(n)))) = \min_{s \in S_s} d(X, \gamma(S')) \tag{10}$$

El segundo punto de vista es el que asocia etiquetas a las transiciones, producidas por la transición del estado actual al estado siguiente y no del último lugar de llegada para un estado, según (11).

$$\beta(U(n),S(n)) = \Gamma(S(n), f(U(n),S(n))) \tag{11}$$

Aunque la diferencia entre las dos es muy sutil, es posible argumentar que estudiando el comportamiento de los estados o de las transiciones es posible determinar una reproducción adecuada en la salida del decodificador.

D. Diseño de Codificadores y Decodificadores

Para el diseño de los codificadores y decodificadores es muy importante definir un procedimiento ordenado y coherente, el proceso esta definido por 4 pasos que son:

- 1) Definición de parámetros básicos.
- 2) Creación del condicional de error, contador para iteraciones y valor inicial de la distorsión.
- 3) Codificar el conjunto de entrenamiento y definir los estados del diccionario, tanto para el codificador como para el decodificador.
- 4) Calcular la distorsión mínima.

Como el proceso pretende encontrar dentro del súper-diccionario la mejora de los diccionarios para cada iteración. El proceso se torna iterativo, para así encontrar el resultado óptimo, el diagrama de flujo correspondiente es presentado en la Fig. 3.

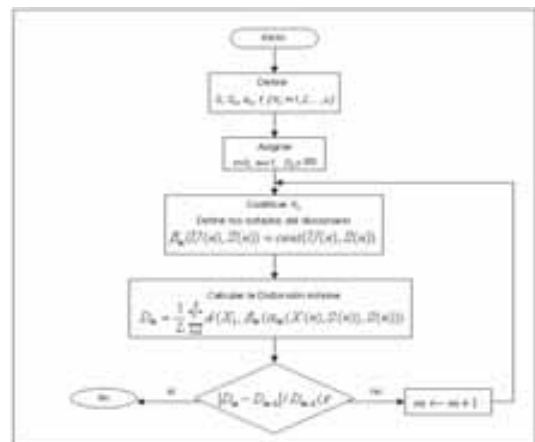


Fig. 3. Diagrama de flujo para el diseño de codificadores y decodificadores.

E. Diseño de la Función Estado Siguiente

Un clasificador de estados para cuantización vectorial puede ser diseñado utilizando el Algoritmo Generalizado de Lloyd (GLA)[8], en este, el diccionario es denotado como el conjunto de vectores códigos asociados a los estados S y esta representado por (12).

$$B = \{C(s); s \in S\} \quad (12)$$

Conocido también como clasificador de diccionarios, este proceso busca encontrar para los vectores de entrada los vecinos mas próximos para el diccionario, el clasificador de estados esta definido por (13).

$$V(X) = \min_{s \in S}^{-1} d(X, C(s)) \quad (13)$$

El diseño del clasificador de estados puede ser elaborado utilizando diferentes técnicas, las más conocidas son:

- 1) *Histograma condicional.*
- 2) *Vecino más próximo.*
- 3) *Diseño Omnisciente.*
- 4) *Diseño de autómatas estocásticos.*

III. OPTIMIZACIÓN DE LA CURVA DE DISTORSIÓN

Para la optimización de la curva de distorsión fue utilizada una matriz de (2, 18000) que representa una palabra de dos silabas, el proceso de cuantización vectorial[9]-[10] fue realizado utilizando dos modelos para así obtener datos comparativos de la mejora de la distorsión en cada uno de los procesos, inicialmente fue utilizado el algoritmo generalizado de Lloyd[11] para realizar la cuantización vectorial de la forma tradicional, fueron utilizadas palabras de longitud variable que comenzaron en 2 bits y terminaron en 8 bits para la codificación, dando como resultado los valores presentados en la figura 4, la condición de parada esta determina por un umbral de mejora igual a $1e^{-3}$ con respecto al valor anterior de distorsión obtenido, este valor es utilizado en los dos algoritmos a fin de conseguir respuestas bajo las mismas condiciones y así determinar el nivel de procesamiento involucrado.

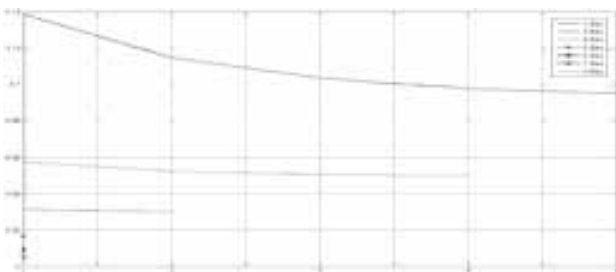


Fig. 4. Curvas de distorsión de número de muestras vs error medio cuadrático para VQ.

Los resultados presentados en la figura 4, demuestran que los valores mínimos de distorsión disminuyen conforme aumenta el número de bits utilizados, lo cual es perfectamente comprensible puesto que al aumentar el número de bits los niveles de cuantización también aumentan.

En la figura 5 son presentadas las curvas obtenidas para el algoritmo de cuantización vectorial de estado finito

desarrollado, debido a la escala utilizada en los últimos resultados es casi imperceptible el cambio producido, sin embargo el criterio de parada sigue siendo el mismo que en el algoritmo anterior.

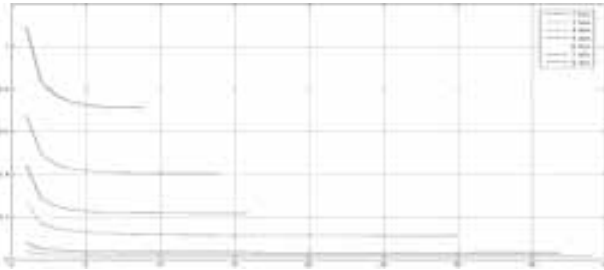


Fig. 5. Curvas de distorsión de muestras vs error medio cuadrático para FSVQ.

Los resultados obtenidos demuestran que son mejorados los valores de distorsión presentados en el algoritmo VQ tradicional, según Tabla II, con esto queda demostrado que la utilización de un algoritmo con características de mejora para cada diccionario utilizado representa notorias ventajas en las tareas de codificación, compresión y tiempo de procesamiento computacional.

TABLA II
COMPARACIÓN DE RESULTADOS OBTENIDOS PARA DISTORSIÓN

Nº DE BITS	VQ	FSVQ
2	712.7×10^{-3}	93.93×10^{-3}
3	405.7×10^{-3}	48×10^{-3}
4	216.8×10^{-3}	29.1×10^{-3}
5	112.3×10^{-3}	15.9×10^{-3}
6	59.2×10^{-3}	8.8×10^{-3}
7	30.85×10^{-3}	5.2×10^{-3}
8	16.16×10^{-3}	3.8×10^{-3}

Otra característica importante de resaltar es notar como el número de iteraciones disminuye notablemente, gracias a los resultados obtenidos es posible afirmar que la utilización de FSVQ garantiza mayores velocidades de procesamiento, tanto así, que para poder conseguir valores de distorsión similares a al VQ tradicional fue necesario trabajar con 256 niveles de codificación lo que supone un número de 8 bits, como característica especial el trabajo alcanzado con el algoritmo FSVQ presento mejoras en todas las configuraciones utilizadas, con un número razonable de iteraciones.

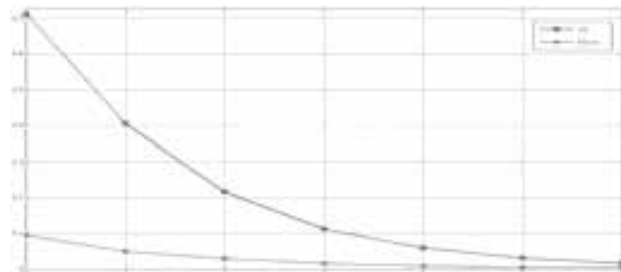


Fig. 6. Desempeño de los modelos utilizados, número de bits vs error medio cuadrático.

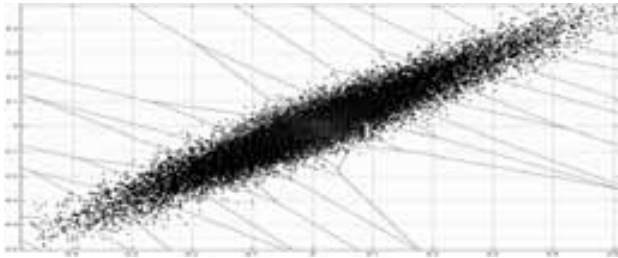


Fig. 7. Diagrama de Voronoi para 8 Bits.

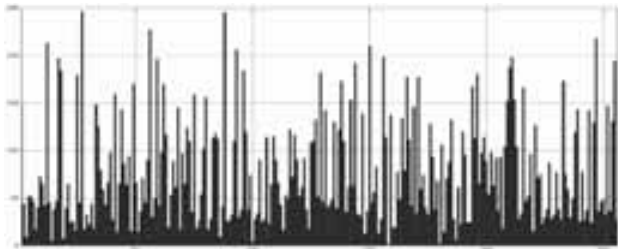


Fig. 8. Comportamiento de los índices para 8 Bits, índices vs número de repeticiones.

En las figuras 7 y 8 son presentados el diagrama de Voronoi y el comportamiento de los índices para 8 bits, en el diagrama de voronoi es apreciable como son creadas las particiones generando 256 niveles. Es así como es posible discriminar de una forma más clara a cual centroide pertenece cada muestra de la señal original, después de ser localizada la región óptima es enviado el coeficiente representativo de este centroide.

Con el fin de tener resultados comprobados de los niveles de error obtenidos utilizando los dos algoritmos, fueron comparados los resultados después del proceso completo, entiéndase por proceso completo, la realización de la codificación y compresión, así como posteriormente la decodificación y descompresión. Utilizado un ecualizador adaptativo LMS¹ se obtuvo el error cuadrático producido entre la señal original y las señales obtenidas después de los procesos de cuantización.

En las figuras 9 y 10 son presentados los niveles de error obtenidos, es notable como para la señal FSVQ el nivel máximo obtenido es casi de la mitad con respecto al obtenido en la señal VQ. Es posible observar que los niveles mas altos en las dos señales se presentan entre las muestras 5000 y 9000, que son básicamente donde se encuentra la mayoría de la información de la primera silaba que tiene acento.

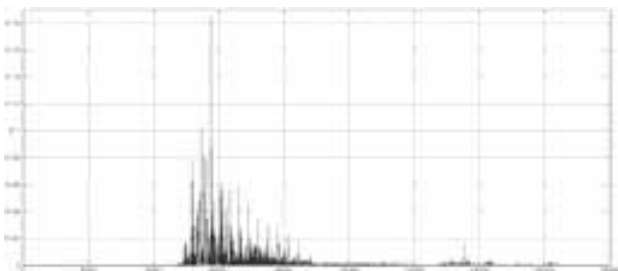


Fig. 9. Nivel de error para VQ utilizando 8 bits, número de muestras vs amplitud.

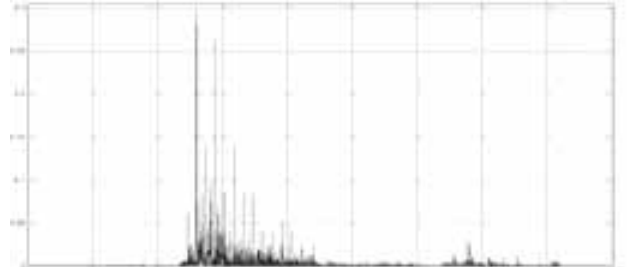


Fig. 10. Nivel de error para FSVQ utilizando 8 Bits, número de muestras vs amplitud.

IV. CONCLUSIONES

Los resultados presentados demuestran que el algoritmo implementado para aplicar cuantización vectorial de estado finito en señales de voz presenta una mejora significativa con respecto a los esquemas tradicionales de cuantización vectorial, además es apreciable que debido a la disminución del número de iteraciones el costo computacional también baja, valor agregado que puede ser usado para definir mejoras en el desempeño, después de ser decodificadas fue apreciable la diferencia de en calidad de las señal cuantizada con VQ y la señal cuantizada con FSVQ, como trabajo futuro se planea desarrollar diferentes configuraciones de estados siguientes para ser probados con diferentes tipos de clasificador y así determinar la mejor combinación dependiendo del tipo de señal trabajada.

REFERENCIAS

- [1] C. H. Valencia, M. S. Dutra, "Modelado del Incremento de Capacidad en Canales Satelitales utilizando Lógica Difusa" Revista Iteckne, 2007, Vol. 4, No 1, pp. 5-11.
- [2] A. Gersho and R. M. Gray, "Vector Quantization and Signal Compression", Ed. Springer, 1992, 760p, ISBN: 0792391810.
- [3] C. Zhu, L. M. Po and Y. Hua, "Optimized feature map finite-state vector quantization for image coding" *Proceedings Visions, Image and Signal Processing*, 2000, pp. 266-270.
- [4] W. Hsien-Chunq, T. Pao-Chin and W. Jia-Shunq, "Three-sided side match finite state vector quantization" *IEEE Conference on Acoustics, Speech and Signal Processing*, 1998, pp. 2669-2672.
- [5] S. Ramakrishnan, K. Rose and A. Gersho, "Constrained-storage vector quantization with a universal codebook" *IEEE Transactions on Image Processing*, 1998, Vol. 7, No 6, pp. 785-793.
- [6] C. Ruey-Feng and H. Yu-Leng, "Finite-state quantization by exploiting interband and intraband correlations for sub band image coding" *IEEE Transactions on Image Processing*, 1996, pp. 374-378.
- [7] T. Kurose and E. Oqawa, "Improvement of finite state vector quantization using classification" *IEEE Conference Record Nuclear Science Symposium and Medical Imaging*, 1995, pp. 1415-1419.
- [8] C. Chanq-Qian, K. Soo-Nqee and P. Sivaprakasapillai, "A modified generalized Lloyd algorithm for VQ codebook design" *Conference Acoustics, Speech and Signal Processing*, 1996, pp. 542-545.
- [9] Q. Shen-En, "Fast vector quantization algorithms based on nearest partition set search" *IEEE Transactions on Image Processing*, 2006, pp. 2422-2430.
- [10] A. Mouchtaris, Y. Agiomyrqiannakis and Y. Stylianou, "Conditional Vector Quantization for Voice Conversion" *IEEE International Conference on Acoustic, Speech and Signal Processing*, 2007, Vol. 4, pp. IV-505-IV-508.
- [11] T. Kaukoranta, P. Franti and O. Nevalainen, "A fast exact GLA based on code vector activity detection" *IEEE Transactions on Image Processing*, 2000, 1337-1342.

¹ LMS – Least-Mean-Square.

Simulación de conversores A/D de alta resolución empleando modulación Delta-Sigma

Prada J., Sedano N., y Vizcaya P., miembro IEEE

Resumen— El desarrollo de conversores de alta resolución (más de 16 bits) no es posible realizarlo empleando técnicas directas (aproximaciones sucesivas o conversores paralelo) por la limitación en la precisión de los elementos empleados y el ruido eléctrico. Para alcanzar altas resoluciones se requiere sobremuestrear la señal y emplear técnicas como *dithering*. Aunque con *dithering* es posible aumentar la resolución proporcionalmente con la tasa de muestreo, la modulación delta-sigma permite alcanzar las mismas resoluciones con tasas de muestreo significativamente menores por conformación del ruido de cuantificación. El presente trabajo describe esta técnica y analiza la influencia en la resolución de los parámetros del modulador, tales como la tasa de muestreo, orden del filtro y niveles del cuantificador básico empleado. Los resultados de esta simulación permitirán plantear la arquitectura de un procesador para implementar un conversor A/D con resolución efectiva de 24 bits, que pueda ser usado como referencia para evaluar componentes de muy baja distorsión y alta resolución.

Palabras Claves: Conversión análogo-digital de alta resolución, modulación delta sigma, conformación del ruido de cuantificación.

I. INTRODUCCION

Con el fin de obtener conversores analógico-digital (A/D) de alta resolución para distintas aplicaciones y para optimización de sistemas electrónicos, se ha desarrollado en este escrito distintas topologías de conversores A/D empleando modulación delta-sigma (DS), ya que con ésta podemos utilizar la conformación de ruido de cuantificación y el sobremuestreo como principales herramientas para obtener una alta resolución, que con otros mecanismos sería imposible realizar. Para alcanzar la máxima resolución posible se realizan cambios en el orden del filtro de lazo del conversor, en los niveles del cuantificador, así como en la tasa de sobremuestreo y la ganancia utilizada, que son parámetros esenciales para lograr buena precisión en la señal de salida.

Manuscrito enviado el 11 de Julio de 2008. Este artículo es el resultado del trabajo de final de curso presentado en la asignatura Sistemas de Comunicación II.

P. Vizcaya es profesor titular de la Facultad de Ingeniería de la P. Universidad Javeriana (pvizcaya@javeriana.edu.co) del área de Comunicaciones y Análisis de Señales.

J. Prada y N. Sedano son estudiantes de la carrera de Ingeniería Electrónica de la Pontificia Universidad Javeriana y actualmente están elaborando la investigación previa para el desarrollo de su trabajo de grado en este tema. Este trabajo fue realizado como parte de la asignatura de Sistemas de Comunicación II.

Para llevar a cabo este proyecto se realizó inicialmente una investigación detallada y posteriormente simulaciones de distintos conversores delta-sigma (DS) de alta resolución empleando MATLAB-SIMULINK, como herramienta base y efectiva. Todo esto con el fin de obtener las bases de una posible arquitectura en hardware de un conversor. Vale la pena aclarar que los conversores se concentrarán en aplicaciones de audio, por lo tanto las señales de entrada de estos conversores trabajarán en el rango de la banda de audio, es decir hasta 20 kHz.

II. MARCO TEÓRICO

Las modulaciones delta y delta-sigma fueron desarrolladas en los años 40 y 60, respectivamente, y se han usado normalmente en aplicaciones de telefonía. Sus limitaciones impidieron su uso en aplicaciones de audio (señales con mejor calidad) hasta que no se desarrollaron en los años 80 las técnicas de procesado digital. La modulación delta-sigma (DS) fue desarrollada con objeto de solventar los problemas que presentaba la modulación delta. En estos codificadores se cuantifica la señal diferencia (delta) entre la muestra de la señal de entrada y la suma (sigma) de las diferencias anteriores [1]. El integrador que en modulación delta se ubica en el receptor, pasa al transmisor en la modulación DS, con lo cual la salida queda proporcional a la entrada.

El modelo del modulador DS de orden n se muestra en la Figura 1.

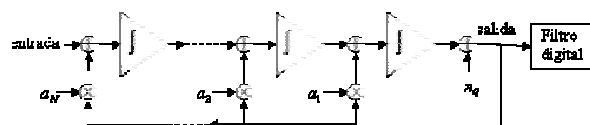


Figura 1. Modulador delta-sigma de orden n

La derivación de la función de transferencia del modulador DS de orden n con solo polos para la señal se puede encontrar usando recurrencia como sigue:

$$\begin{aligned}
H_{señal}^{(1)}(s) &= \frac{1}{s\tau_1 + a_1} \\
H_{señal}^{(n-1)}(s) &= \frac{1}{s^{n-1}\tau_1 \dots \tau_{n-1} + s^{n-2}\tau_2 \dots \tau_{n-1}a_1 + \dots + a_{n-1}} \\
H_{señal}^{(n)}(s) &= \frac{H_{señal}^{(n-1)}(s)}{s\tau_n + a_n H_{señal}^{(n-1)}(s)} \\
&= \frac{1}{s^n \tau_1 \dots \tau_n + s^{n-1} \tau_2 \dots \tau_n a_1 + \dots + a_n}
\end{aligned} \quad (1)$$

Esta función de transferencia corresponde a un filtro pasabajos. La función de transferencia para el ruido se puede deducir en forma similar como:

$$H_{n_q}^{(n)}(s) = \frac{s^n \tau_1 \dots \tau_n}{s^n \tau_1 \dots \tau_n + s^{n-1} \tau_2 \dots \tau_n a_1 + \dots + a_n} \quad (2)$$

Esta función de transferencia corresponde a un filtro pasaaltos. Se pueden escoger todas las constantes de tiempo iguales, digamos $\tau_i = \tau$, y determinar la respuesta del filtro para la señal con las constantes a_i y τ :

$$H_{señal}^{(n)}(s) = \frac{1}{s^n \tau^n + s^{n-1} \tau^{n-1} a_1 + \dots + a_n} \quad (3)$$

La respuesta del filtro se puede ajustar para que se tenga la de un Butterworth pasabajos y se debe escoger el τ tan alto como se pueda, para que la frecuencia de corte del filtro sea lo más alta posible y así la atenuación del filtro pasaaltos a la frecuencia de interés sea lo más alta posible. Con un cuantificador multinivel el tiempo de respuesta es importante pues la señal debe estar oscilando entre 2 niveles al menos para que el ruido tenga efectivamente componentes en alta frecuencia (conformación del ruido).

Diseño con MATLAB: la función *butter* se usó para determinar los coeficientes del filtro análogo. La potencia del ruido se define como:

$$\begin{aligned}
N &= \int_{-\Omega_c/2}^{\Omega_c/2} \frac{\Delta^2}{12 \times 2\pi f_s} \left| H_{n_q}^{(n)}(\Omega) H_{FiltroDigital}(\Omega) \right|^2 d\Omega \\
&\approx \frac{\Delta^2}{12 \times 2\pi f_s} \int_{-2\pi B}^{2\pi B} \left| \frac{s^N \tau^N}{s^N \tau^N + s^{N-1} \tau^{N-1} a_1 + \dots + s\tau a_{N-1} + a_N} \right|_{s=j\Omega}^2 d\Omega
\end{aligned} \quad (4)$$

En la anterior expresión se ha usado el hecho de que la respuesta del filtro digital de reconstrucción es un pasabajos con frecuencia de corte igual al ancho de banda B de interés para la señal y atenuación muy alta para componentes por encima de esta frecuencia. Entonces la potencia del ruido se puede aproximar a

$$N \approx \int_{-2\pi B}^{2\pi B} \frac{\Delta^2}{12 \times 2\pi f_s} \frac{(\Omega \tau)^{2n}}{a_n^2} d\Omega$$

Haciendo $\Omega' = \Omega \tau$

$$\begin{aligned}
N &= \int_{-2\pi B\tau}^{2\pi B\tau} \frac{\Delta^2}{12 \times 2\pi f_s} \frac{(\Omega')^{2n}}{a_n^2} d\Omega' \\
N &= \frac{\Delta^2}{12 \times 2\pi f_s \tau} \frac{(\Omega')^{2n+1}}{(2n+1)a_n^2} \Big|_{-2\pi B\tau}^{2\pi B\tau} \\
N &= \frac{\Delta^2}{12 \times 2\pi f_s \tau} \frac{2(2\pi B\tau)^{2n+1}}{(2n+1)a_n^2}
\end{aligned} \quad (5)$$

Esta expresión muestra la dependencia de la potencia del ruido por distorsión de τ^{2n} . Δ es el paso de cuantificación. La potencia de la señal, en nuestro caso, corresponde a la potencia de la señal sinusoidal de prueba:

$$S = \frac{A_{in}^2}{2} \quad (6)$$

Por lo tanto, la relación señal a distorsión se puede aproximar a:

$$\begin{aligned}
SNR(dB) &= 10 \log_{10} \left(\frac{S}{N} \right) \\
SNR(dB) &= 10 \log_{10} \left(\frac{(2n+1)A_{in}^2 6\pi f_a \tau}{\Delta^2 (2\pi B\tau)^{2n+1}} \right)
\end{aligned} \quad (7)$$

donde A_{in} es la amplitud de entrada, Δ es igual a $2/(L-1)$, siendo L en número de niveles del cuantificador, a_n es 1, B es el ancho de banda a trabajar que se encontrará fijo en 20kHz (banda de audio), n es el orden del filtro de lazo de conformación de ruido y f_a es la frecuencia de muestreo ($f_a = Rf_s$, en donde f_s es la tasa de Nyquist, en este caso 48kHz, y R es el factor de sobremuestreo). Ya teniendo la relación señal a distorsión SNR , una forma de medir la resolución del conversor es a través del número de bits efectivos a la salida n_{bits} [9], equivalente a

$$n_{bits} = \frac{SNR(dB) - 1.76}{6.02} \quad (8)$$

III. DESARROLLO Y SIMULACIONES

En una modulación delta-sigma (DS) de primer orden o de integración simple, tal como se muestra en la Figura 2, la entrada al cuantificador es la integral de la diferencia entre la señal de entrada y salida cuantificada. El integrador realiza un filtrado paso bajo en la señal diferencia, proporcionando de esta forma un lazo de baja frecuencia en el cuantificador. Esta realimentación produce una reducción del ruido de cuantificación en bajas frecuencias, debido a que desplaza el ruido de cuantificación a frecuencias superiores (ver Figura 9).

A diferencia de las codificaciones PCM y DM (Delta), el ruido no es blanco (espectro plano), sino que está conformado espectralmente con un filtro paso alto de primer orden. Así como se puede realizar una modulación DS de primer orden, es decir con filtro de lazo para conformación del ruido de cuantificación de primer orden, se pueden realizar filtros de orden superior, sin embargo a medida que aumenta el orden del conversor aumenta la complejidad y las inestabilidades en

el sistema. Es por eso que hemos llegado en nuestra investigación a realizar convertidores de tercer y cuarto orden, que inclusive presentan inestabilidades en algunos casos, como al aumentar o disminuir demasiado la ganancia del filtro o aumentar demasiado el factor de sobremuestreo.

En la Figura 2, en la Figura 3, en la Figura 4 y en la Figura 5 se muestran los esquemáticos en Simulink de los convertidores desarrollados de 1^{er}, 2^{do}, 3^{er} y 4^{to} orden respectivamente [2]. Cada esquemático a simular contiene un filtro RLS para la medición de la relación señal a distorsión SNR, un bloque para rediseñar el cuantificador, aumentando o disminuyendo los niveles, un bloque que representa el filtro digital a la salida del convertor que es importante para regenerar la señal original, filtrar las frecuencias altas del ruido de cuantificación y realizar un diezmado, reduciendo la frecuencia de muestreo [3]. Éste filtro tendrá una frecuencia de paso fija hasta 20 kHz en todas las simulaciones.

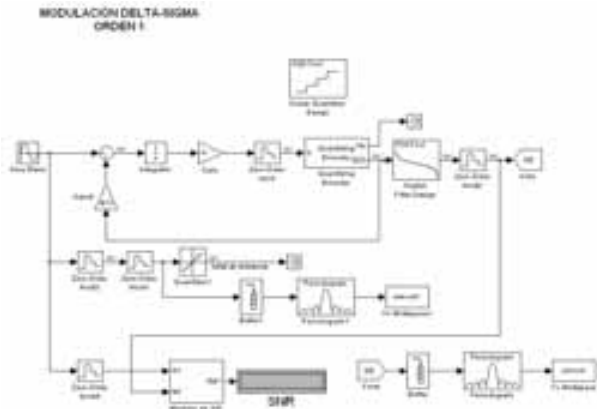


Figura 2. Conversor delta-sigma de primer orden

Además del bloque del filtro digital, los esquemáticos contienen los lazos de realimentación según el orden del convertor con sus respectivos integradores, muestreo y retención de orden 0 y ganancias, así como visores para observar los distintos espectros y señales en tiempo, y así observar el comportamiento del convertor.

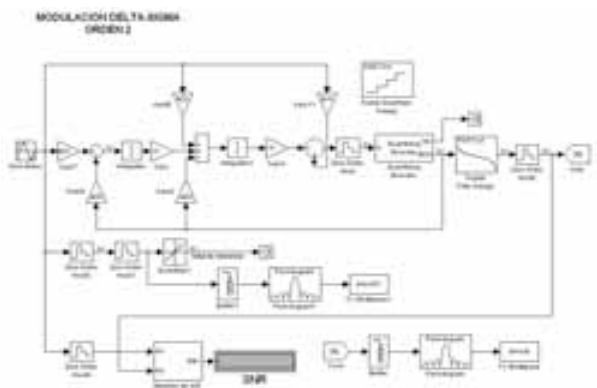


Figura 3. Conversor delta-sigma de segundo orden

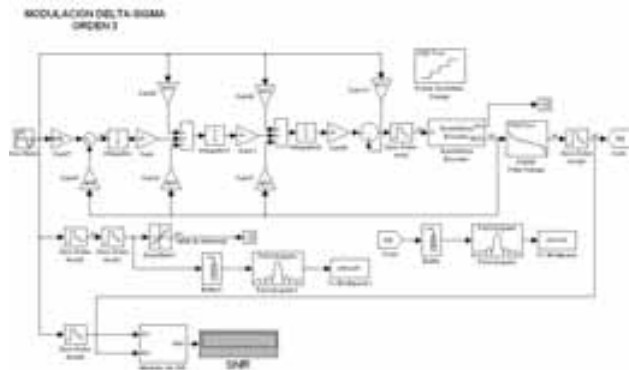


Figura 4. Conversor delta-sigma de tercer orden

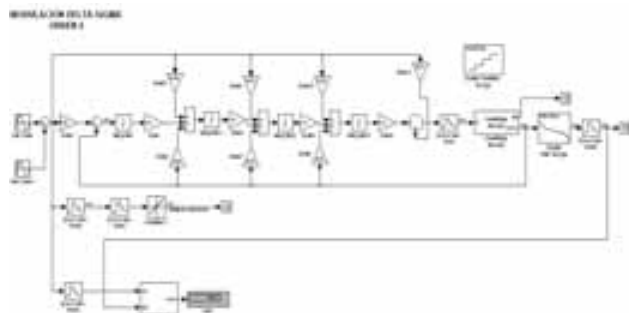


Figura 5. Conversor delta-sigma de cuarto orden

A. Cuantificación

Una señal continua, tal como la voz, tiene un intervalo continuo de amplitudes, y por tanto sus muestras tienen un intervalo continuo de amplitud. Como se desea generar una señal digital, no basta solo con muestrear en el tiempo, sino que es necesario cuantificar. El oído humano o el ojo, como un receptor final, puede detectar únicamente diferencias de intensidad finita, por lo que no es necesario transmitir las amplitudes exactas de las muestras, basta con aproximar mediante una señal construida a partir de amplitudes discretas elegidas de un conjunto disponible sobre una base de error mínimo [9]. Para esto se varían los niveles del cuantificador, pasando de una cuantificación de un bit a multibit, por lo que es de esperar que a mayor número de pasos de cuantificación menor será el ruido de cuantificación. En la Tabla 1 (Figura 6) se varia el número de niveles L del cuantificador entre 2, 4, 8 y 16, se deja fija un factor de sobremuestreo de 64, con una tasa de muestreo f_s de Nyquist de 48000Hz, también se deja fija la amplitud y la frecuencia de entrada en 0.9 unidades y en 1009Hz respectivamente, f_a es 3072 kHz.

Tabla 1. Resultados prácticos y teóricos de la SNR variando los niveles de cuantificación

Orden filtro n	L	Δ	τ	SNR simulado (dB)	SNR Teórico (dB)
1	16	2/15	1/3100000	75.58	75.83
1	8	2/7	1/3200000	69.41	69.49
1	4	2/3	1/3400000	62.8	62.65

Orden filtro n	L	Δ	τ	SNR simulado (dB)	SNR Teórico (dB)
1	2	2	1/2500000	56.56	50.44
2	16	2/15	1/1600000	93.95	94.41
2	8	2/7	1/1800000	90.19	89.83
2	4	2/3	1/1300000	77.52	76.82
2	2	2	1/1300000	73.48	67.27
3	16	2/15	1/1300000	112.4	112.55
3	8	2/7	1/1300000	106.3	105.93
3	4	2/3	1/700000	80.68	82.44
3	2	2	1/600000	73.94	68.88
4	16	2/15	1/1000000	125.1	124.82
4	8	2/7	1/1000000	118.3	118.20
4	4	2/3	1/530000	88.59	88.78
4	2	2	1/360000	77.29	65.80

SNR

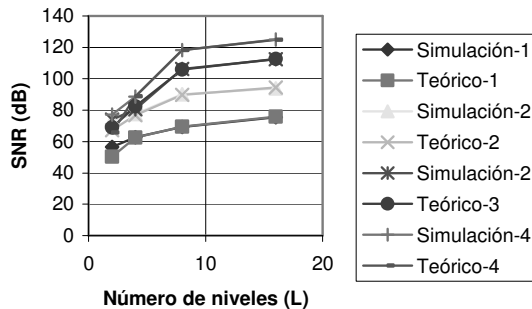


Figura 6. Relación SNR con limitación de la oscilación a 2 niveles.

El valor óptimo de τ o de la ganancia del filtro $1/\tau$ se puede modificar para que el convertor entregue la mayor SNR posible o para que se obtenga una señal de salida que no tenga saltos de más de un nivel a la vez (ver Figura 6, representada por los resultados de la Tabla 1). La Figura 7 muestra la máxima SNR obtenida al variar los distintos parámetros, obteniendo una diferencia de 70 dB aproximadamente entre la conversión de primer y cuarto orden, que equivale, al reemplazar en la ecuación (8), a una diferencia de 13 bits efectivos. Lo anterior teniendo en cuenta cuantificadores de 16 niveles, ya que como se muestra la Figura 7 con 2 y 4 niveles de cuantificador, ocurre una disminución de los conversores de tercer y cuarto orden debido a una inestabilidad basada en la complejidad del sistema en estos ordenes, manteniendo un valor de SNR similar al obtenido en primer y segundo orden. La máxima SNR obtenida en estas simulaciones fue 146.0 dB, en un convertor de cuarto orden empleando 16 niveles de cuantificación y con un factor de sobremuestreo de 64, que equivale a 23.96 bits efectivos a la salida. La Figura 8 presenta el espectro de salida del anterior resultado para una señal de entrada de 7001 Hz.

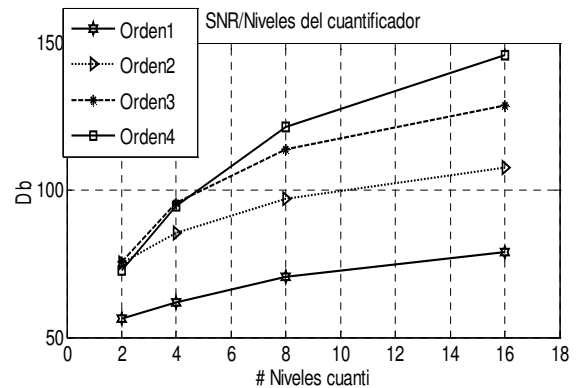


Figura 7. Relación señal a ruido según el número de niveles del cuantificador empleados.

B. Conformación del ruido de cuantificación

En la Figura 9 se muestra el espectro del ruido de cuantificación, en la cual se aprecia la conformación producida por el filtro del modulador DS. La pendiente de la caída del ruido llega a ser 80 dB en una década (entre 2KHz y 20 KHz) como era de esperarse en un filtro de cuarto orden.

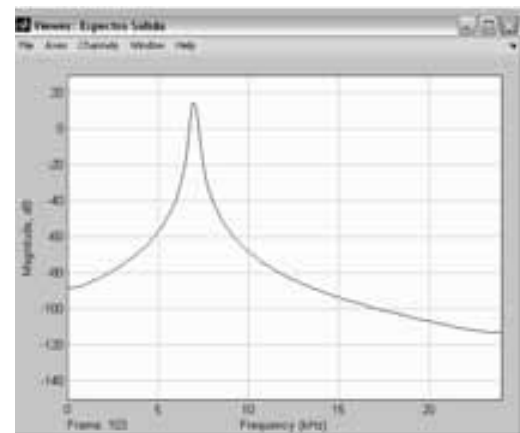


Figura 8. Espectro de salida para una señal con frecuencia de 7001 Hz

C. Sobremuestreo

El empleo de un factor de sobremuestreo R reduce enormemente el nivel de ruido de cuantificación en la banda deseada, ya que el ruido se distribuye desde 0 a $f_a/2$ Hz, donde $f_a=Rf_s$ es la tasa de sobremuestreo utilizada. Para los resultados en la Tabla 1 se emplea un factor de sobremuestreo de 64; en la Tabla 2 (ver Figura 10) se varía ahora éste factor a 32, teniendo así una frecuencia de sobremuestreo f_a de 1.536 MHz.

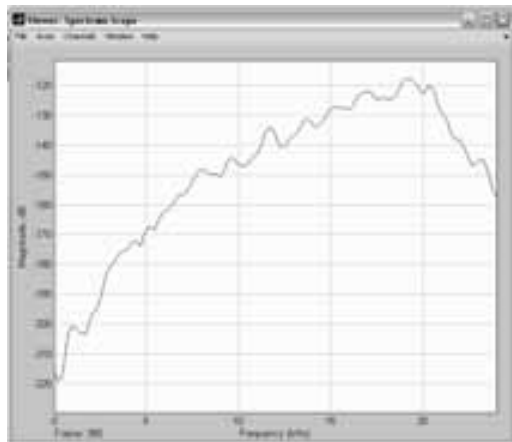


Figura 9. Conformación del ruido de cuantificación para el conversor delta-sigma de cuarto orden después del filtro digital

Tabla 2. Resultados prácticos de la SNR variando los niveles de cuantificación y el sobremuestreo a un factor de 32

Orden filtro	Niveles cuantif.	1/τ	S/N (dB)	Bits Efectivos
1	16	2500000	70.23	11.37
1	8	2300000	62.44	10.07
1	4	2000000	54.41	8.74
1	2	5000000	47.88	7.66
2	16	1900000	92.19	15.02
2	8	1700000	82.69	13.44
2	4	1500000	72.38	11.73
2	2	1100000	60.35	9.73
3	16	1300000	107.7	17.59
3	8	1000000	93.30	15.20
3	4	650000	76.46	12.40
3	2	300000	54.07	8.68
4	16	1900000	116.9	19.12
4	8	600000	95.38	15.55
4	4	400000	78.2	12.69
4	2	175000	50.96	8.17

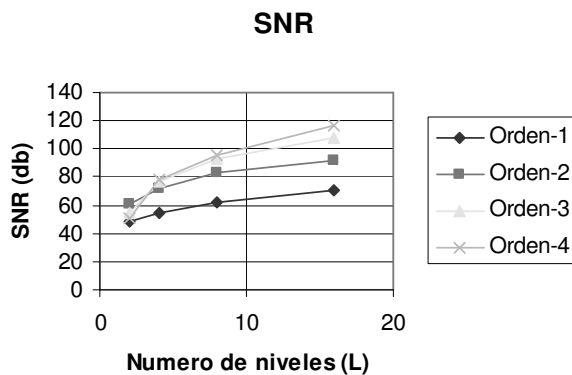


Figura 10. Relación señal a ruido para un factor de sobremuestreo de 32.

D. Señal de entrada

Entre otras simulaciones realizadas, se modificaron los parámetros de la frecuencia y la amplitud de la señal de entrada, para observar cómo afectan el cambio de éste en la resolución de conversor. Para la Tabla 3 de resultados se varía la amplitud de la señal de entrada con caídas de 3dB, se deja fija la frecuencia de la señal en 1009Hz y el número de niveles L del cuantificador en 16.

Tabla 3. Resultados prácticos de la SNR variando caídas en la amplitud de la señal de entrada

Orden filtro	A in	1/τ	S/N (dB)	Bits Efectivos
1	0.7071	5000000	73.73	11.95
1	0.5	5000000	72.28	11.71
1	0.3533	5000000	70.01	11.33
1	0.25	5000000	65.34	10.56
2	0.7071	2600000	106.3	17.36
2	0.5	2600000	103.2	16.85
2	0.3533	2600000	100.3	16.36
2	0.25	2600000	97.52	15.90
3	0.7071	2600000	128.1	20.98
3	0.5	2600000	124.9	20.45
3	0.3533	2600000	121.0	19.80
3	0.25	2600000	118.8	19.44
4	0.7071	1900000	143.9	23.61
4	0.5	1900000	140.9	23.11
4	0.3533	1900000	137.9	22.61
4	0.25	1900000	134.7	22.08

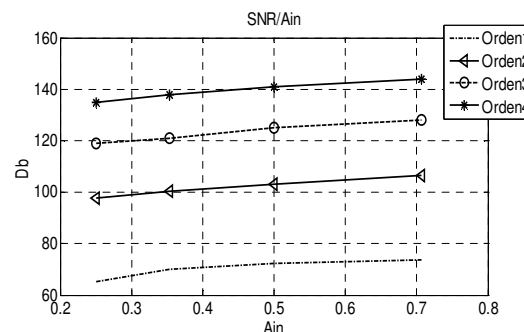


Figura 11. Relación señal a distorsión variando la amplitud de la señal de entrada

Al variar la frecuencia de entrada en los distintos conversores se obtiene una SNR similar en todos los casos, por lo que se comprueba que el número de bits efectivo a la salida n_{bits} no depende de la frecuencia de la señal de entrada. En cambio, en la Tabla 3 (resultados representado en la Figura 11), la disminución de amplitud afecta la SNR; como ejemplo se tiene que de 0.7071 a 0.25 en cada uno de los órdenes de conversión DS existe una caída aproximadamente 7 u 8 dB, equivalentes a un bit efectivo.

IV. CONCLUSIONES

Como se había establecido, se deseaba realizar simulaciones de conversores delta-sigma de alta resolución con lo cual se cumplió, ya que la máxima SNR obtenida fue 146.0 dB equivalentes a 23.96 bits efectivos, esto para un conversor DS de cuarto orden con un factor de sobremuestreo de 64 y con 16 niveles de cuantificador. Por debajo de este, se ubican otros resultados de acuerdo a las necesidades o resoluciones que se deseen según su aplicación.

Se desarrollaron en definitiva conversores que varían desde 7 a 24 bits efectivos a la salida, que implican modelos de menor o mayor complejidad, y de esta forma menor o mayor requerimientos a la hora de implementar el conversor. Cuanto mayor es el orden del conversor mayor es la susceptibilidad del mismo a inestabilizarse.

Utilizando un cuantificador de dos niveles la relación señal a ruido no depende del valor de τ a excepción del conversor de orden cuarto, en el cual si se verifica una relación directa entre el τ y la SNR.

El cálculo teórico de SNR presentado en la Tabla 1 resulta menor que la máxima SNR obtenida mediante las simulaciones porque la ecuación (7) no contempla la mejora, que pudimos verificar en simulación, esto se logra cuando la señal a la salida del cuantificador no sólo oscila entre dos niveles, sino que comienza a oscilar entre dos y tres niveles; el problema con esta mejora es que va llevando el conversor a estado inestable. Por lo tanto, la ecuación (7) utilizada para calcular el SNR no contempla el caso en el que se usa un cuantificador de solo dos niveles, pero en los otros casos analizados, es decir para 4, 8 y 16 niveles de cuantificación el cálculo teórico se aproxima de manera bastante satisfactoria a la simulada.

REFERENCIAS

- [1] K. C. Pohlmann, *Principios de Audio Digital*. 4 Edición. Mcgraw hill, pp. 631-668.
- [2] R. Schreier and G. C. Temes, *Understanding Delta-Sigma Data Converters*, John Wiley & Sons, New York, 2004
- [3] S. Park, *Principles of Sigma-Delta Modulation for Analog to Digital Converters*, Motorola Digital Signal Processors, Apr8, Rev.1. Strategic Application.
- [4] G. Mulliken, F. Adil, G. Cauwenberghs and R. Genov, *Delta-Sigma Algorithmic Analog-to-digital conversion*. Base datos IEEE.
- [5] S. Vijjapu, Tesis: *RC Implementation of an audio frequency band butterworth MASH Delta-Sigma analog to digital Data converter*. Bachelor of Engineering, Jawaharlal Technological University, 2002.
- [6] A. Yasuda, H. Tanimoto and T. Iida, *A Third-Order delta sigma modulator using second-order Noise-shaping Dynamic Element Matching*. IEEE journal of solid-state circuits, VOL. 33, No12, December 1998.
- [7] C. Binan, *A 20 Bit 25 KHz Delta Sigma A/D Converter utilizing frequency-shaped Chopper Stabilization scheme*. Burr Brown Corp. IEEE 2000 Custom integrated circuits conference.
- [8] S. Lindfors, M. Lansirinne, T. Lindeman, K. Halonen, *On the desing of 2ND order multibit delta sigma modulators*, Helsinki University of Technology, Base Datos IEEE 1999.
- [9] S. Haykin, *Sistemas de Comunicación*. Editorial Limusa Wiley, Edicion 2.1, 2006. Pp 218-223.

Pedro R. Vizcaya (BSc EE, P. Universidad Javeriana, Bogotá, Colombia, 1980, MSc, 1986 y PhD, 1998, Rensselaer Polytechnic Institute, Troy, NY) es profesor titular de la Facultad de Ingeniería de la P. Universidad Javeriana, con la cual se encuentra vinculado desde 1980. Es profesor del área de

Análisis de Señales y Comunicaciones. Sus intereses de investigación están procesamiento de señales, reconocimiento de patrones y teoría de las comunicaciones.

Juan P. Prada Estudiante de décimo semestre de ingeniería electrónica de la P. Universidad Javeriana Bogotá, Colombia. Interesado en la teoría de las comunicaciones y en formas de comunicación óptica.

Néstor A. Sedano es estudiante de décimo semestre de Ingeniería Electrónica en la Pontificia Universidad Javeriana de Bogotá. Está involucrado actualmente en Comunicación digital, redes ópticas de transporte y nanoelectrónica.

Control visual basado en líneas utilizando un sistema de visión con dos cámaras

L. Angel, Member, IEEE

Abstract—Este artículo describe un esquema de Control Visual Basado en Imagen para controlar un robot de 3 grados de libertad (g.d.l.), usando un sistema de dos cámaras en configuración cámara fija. Se proponen dos métodos de estimar el Jacobiano de la imagen on line utilizando líneas como características visuales. La primera de ellas toma en cuenta las ventajas de la representación de Plücker, junto con las propiedades del sistema estéreo a fin de obtener información de profundidad de las líneas 3D observadas. La segunda considera propiedades que miden las características directamente en el plano de la imagen. El sistema de control emplea una ley de control proporcional basada en la pseudoinversa del Jacobiano de la imagen. Los resultados experimentales obtenidos validan los métodos propuestos. Estas pruebas han sido extendidas al Jacobiano analítico de la imagen utilizado en la literatura.

Index Terms—Control Visual, Jacobiano de la imagen.

I. INTRODUCCION

CONTROL Visual es el nombre de todas las técnicas que utilizan la información entregada por sensores visuales para controlar un robot o un sistema articular. Existen dos principales estrategias de Control Visual: Control Visual Basado en Posición (PBVS) y Control Visual Basado en Imagen (IBVS) [10] [11]. En PBVS, tanto la señal de error como la ley de control están definidos en coordenadas cartesianas en función de la posición y orientación actual y deseada del objeto a controlar. En IBVS, el error y la ley de control están expresados directamente en el espacio de características de la imagen en función de las características actuales y deseadas del objeto a controlar.

El IBVS está asociado al cálculo del llamado Jacobiano de la imagen que relaciona linealmente cambios en características de la imagen y cambios en parámetros del espacio de trabajo o cambios en articulaciones del robot [5] [12] [15]. El Jacobiano de la imagen puede ser obtenido ya sea por un cálculo analítico o por una estimación.

El Jacobiano analítico [6] depende de los parámetros intrínsecos de la calibración de la cámara, la estimación de la profundidad Z , de la posición de la característica en el espacio de trabajo del robot, la calibración cinemática de la cámara, y la calibración cinemática del robot. Todo esto, hace del Jacobiano analítico muy propenso a errores. La mayoría de

trabajos previos en control visual suponen conocida la estructura y los parámetros del sistema, u obtenidos a través de un proceso de identificación fuera de línea. Un esquema de control con identificación de parámetros fuera de línea, no es robusto ante perturbaciones, cambios de parámetros o entornos desconocidos.

Pocos trabajos han sido dedicados a Control Visual a partir de líneas, la principal razón es la dificultad de manejar las características de las líneas en la imagen, las cuales representan en geometría proyectiva la proyección de un plano 3D que pasa por el centro óptico de la cámara (Fig. 1).

Aplicaciones en sistemas con una sola cámara se tiene en [6], donde la tarea es posicionar la cámara con respecto a una autopista representada por tres líneas paralelas que están en un plano, el Jacobiano de la imagen es calculado considerando constante la altura a la que se encuentra la cámara sobre la autopista. En [1] se representa una línea en base a coordenadas de Plücker para desacoplar la ley de control en dos partes: alineación de la orientación y alineación del plano de interpretación. En [14] un globo aeroestático robótico con una cámara a bordo es controlado visualmente de manera que la línea central de una autopista quede verticalmente en el centro de la imagen y las líneas laterales queden con una inclinación simétrica a ambos lados. En esta aplicación se utiliza un controlador LQR.

Aplicaciones con dos vistas se tiene en [7] donde una línea 3D es alineada con otra por medio de un esquema de control híbrido y desacoplado a partir de la homografía entre características de las líneas polares en las imágenes de las posiciones actual y deseada de una cámara catadióptrica. En [8], las coordenadas de Plücker son usadas para representar una línea y se lleva a cabo una tarea de alineamiento como dos tareas simultáneas de alineamiento punto a línea.

A diferencia de realizar tareas de Control Visual basadas en el Jacobiano analítico de la imagen, en este artículo se ha elegido estimar el Jacobiano de la imagen través del método de los últimos movimientos linealmente realizados [2] [5] [15]. Esto da robustez y flexibilidad al sistema especialmente en aplicaciones en entornos desconocidos o cambiantes.

Este artículo está organizado de la siguiente manera: En la sección 2 se describe el Jacobiano de la imagen. En la sección 3 se muestra la representación de líneas. Los métodos propuestos y el Jacobiano analítico se describen en las secciones 4 y 5 respectivamente. La ley de control se presenta en la sección 6. En la sección 7 se describe el equipo de pruebas y los experimentos llevados a cabo, y finalmente la sección 8 presenta las conclusiones obtenidas.

L. Angel is with the Universidad Pontificia Bolivariana, Bucaramanga, Colombia, email: lasilva@upbga.edu.co

II. JACOBIANO DE LA IMAGEN

Sea $\mathbf{r}=[r_1 \ r_2 \ \dots \ r_p]^T$ el vector p -dimensional que representa la posición del efector final en un sistema de coordenadas cartesianas. Sea $\mathbf{q}=[q_1 \ q_2 \ \dots \ q_n]^T$ el vector n -dimensional que representa la posición de las articulaciones del robot. Sea $\mathbf{s}=[s_1 \ s_2 \ \dots \ s_m]^T$ el vector m -dimensional que representa las características de la imagen (cualesquiera, ya sea por ejemplo puntos, líneas, momentos de imagen, etc).

La relación entre las velocidades de las articulaciones del robot $\dot{\mathbf{q}}$ y sus correspondientes velocidades en el espacio de la tarea $\dot{\mathbf{r}}$ es representada en términos del Jacobiano del robot \mathbf{J}_{rq} como $\dot{\mathbf{r}}=\mathbf{J}_{rq}\dot{\mathbf{q}}$. La relación entre la velocidad de características de la imagen $\dot{\mathbf{s}}$ y velocidades en el espacio de la tarea está dada por $\dot{\mathbf{s}}=\mathbf{J}_{sr}\dot{\mathbf{r}}$.

La velocidad de las características de la imagen puede ser directamente relacionada a la velocidad de las articulaciones en términos de un Jacobiano compuesto conocido como el Jacobiano Visual-motor [5] [15]:

$$\dot{\mathbf{s}}=\mathbf{J}_{sq}\dot{\mathbf{q}}=\begin{bmatrix} \frac{\partial s_1}{\partial q_1} & \dots & \frac{\partial s_1}{\partial q_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial s_m}{\partial q_1} & \dots & \frac{\partial s_m}{\partial q_n} \end{bmatrix} \dot{\mathbf{q}}, \text{ con } \mathbf{J}_{sq}=\mathbf{J}_{sr}\mathbf{J}_{rq}=\mathbf{J} \quad (1)$$

A. Estimación del Jacobiano

Existen varios métodos para estimar el jacobiano descrito por (1) que van desde los métodos recursivos a los no recursivos [13]. En todos ellos es necesario un Jacobiano inicial que se obtiene realizando una serie de movimientos básicos linealmente independientes. De entre ellos se ha elegido la estimación basada en los últimos movimientos [15] debido a la facilidad de implementación, además de requerir un bajo costo de computación. Si los cambios en características de la imagen y los cambios de posición de las articulaciones en el momento k están representados por $\Delta \mathbf{s}_k=\mathbf{s}_k-\mathbf{s}_{k-1}$ y por $\Delta \mathbf{q}_k=\mathbf{q}_k-\mathbf{q}_{k-1}$ respectivamente, y además se asume que el jacobiano es constante para N movimientos, entonces, éste puede ser estimado como la matriz que simultáneamente satisface N ó más movimientos:

$$\begin{bmatrix} \Delta \mathbf{s}_{k-N+1}^T \\ \vdots \\ \Delta \mathbf{s}_k^T \end{bmatrix}=\begin{bmatrix} \Delta \mathbf{q}_{k-N+1}^T \\ \vdots \\ \Delta \mathbf{q}_k^T \end{bmatrix} \mathbf{J}_k^T \quad (2)$$

donde \mathbf{J}_k es el Jacobiano de la imagen en el momento k .

III. REPRESENTACIÓN DE LÍNEAS

Una línea 3D puede ser adecuadamente representada usando las coordenadas binormalizadas de Plücker [1] [8]. Esta

representación está compuesta de una dupla (\mathbf{n}, \mathbf{u}) , donde \mathbf{n} es un vector perpendicular al plano formado por la línea 3D y el centro óptico de la cámara (llamado plano de interpretación), y \mathbf{u} es la dirección 3D de la línea (Fig. 1).

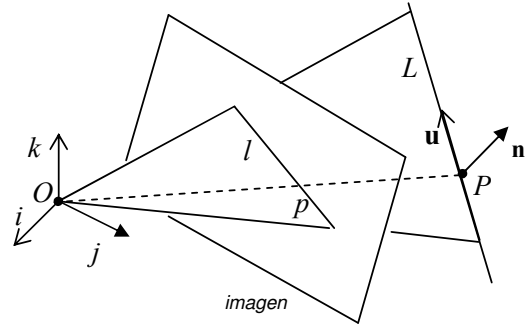


Fig. 1. Interpretación geométrica de las coordenadas binormalizadas de Plücker.

Un plano en el espacio 3D está representado por:

$$aX+bY+cZ=d \quad \text{ó} \quad \mathbf{n}^T \mathbf{X}=d \quad (1)$$

donde $\mathbf{X}=(X,Y,Z)^T$ es un punto en el espacio 3D, $\mathbf{n}=(a,b,c)^T$ es un vector normal al plano y d es la distancia del plano al origen del sistema de coordenadas. Si el plano pasa por el centro óptico de la cámara, $d=0$, (plano de interpretación), por lo que se tiene: $aX+bY+cZ=0$ (Fig. 1).

Dividiendo por Z se obtiene: $ax+by+c=0$ ó

$$(a,b,c)^T(x,y,1)=0 \quad (2)$$

que es la ecuación de la línea $(a,b,c)^T$ en el plano de la imagen y que a la vez representa la intersección entre el plano de interpretación y el plano de la imagen, siendo x e y coordenadas normalizadas de la imagen. Los elementos de esta línea en la imagen son los elementos del vector normal \mathbf{n} al plano de interpretación.

Usando solamente una cámara, no es posible controlar la posición de la línea 3D que descansa en el plano de interpretación sin la ayuda de ningún otro dispositivo. Pero si se utilizan dos cámaras, la segunda ayudará a determinar la posición y orientación de la línea 3D como la intersección de ambos planos de interpretación (Fig. 2).

La representación canónica de la línea en el plano de la imagen es $ax+by+c=0$, sin embargo, esta representación tiene el problema de que tanto la pendiente como el término independiente se hacen infinitos a medida que la línea se hace vertical. Por lo que es más conveniente definir la línea utilizando la llamada representación normal de la línea que posee parámetros (θ, ρ) [14]:

$$x \cos \theta + y \sin \theta = \rho \quad (3)$$

donde θ y ρ son características de la imagen definidas como

$$\theta = \arctan\left(\frac{b}{a}\right) \text{ y } \rho = -\frac{c}{\sqrt{a^2 + b^2}}.$$

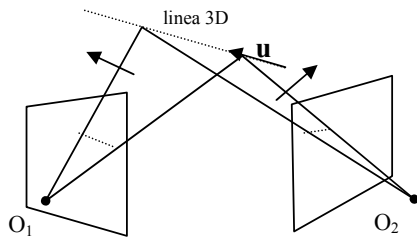


Fig. 2. Dos vistas de una línea 3D y sus respectivos planos de interpretación.

IV. MÉTODOS PROPUESTOS

La especificación de la tarea puede ser expresada de la siguiente manera: dado un juego de líneas en una posición inicial, el sistema debe ser capaz de alinearlas con sus correspondientes líneas en una posición deseada. Para hacer esto, se han implementado dos métodos para controlar el sistema mediante realimentación visual donde los parámetros de las líneas son considerados como características de la imagen.

A. Método NORM

Se sabe que en geometría proyectiva las direcciones 3D tienen solamente dos grados de libertad, si el vector normal al plano de interpretación $\mathbf{n} = [n_x \ n_y \ n_z]^T$ se encuentra normalizado, sus elementos representan los cosenos directores con referencia al sistema de la cámara. Dos de ellos son suficientes para representar la dirección del vector, y por lo tanto son tomados como características. Si se contara con una sola cámara sería imposible recuperar información 3D de la línea pues se sabría que la línea yace en el plano de interpretación pero no se sabría nada acerca de su orientación en el mismo (Fig. 1). Si se considera una segunda cámara, la orientación 3D puede ser detectada como la intersección de ambos planos de interpretación (Fig. 2). Considerando ambas cámaras se tienen cuatro parámetros para controlar el sistema, y el vector construido a partir de ellos tiene la forma:

$$\mathbf{s} = [n_{y1} \ n_{z1} \ n_{y2} \ n_{z2}]^T \quad (4)$$

donde los subíndices 1 y 2 hacen referencia a la primera y a la segunda cámara. El Jacobiano estimado es una matriz $4 \times n$ donde n es el número de grados de libertad del sistema articular. Si se consideran más líneas simplemente se apilan los respectivos Jacobianos, así por ejemplo si se tienen M líneas se tendrá un Jacobiano de dimensión $4M \times n$.

El Jacobiano de la imagen es estimado con el método de los N últimos movimientos realizados [15]. Con varios

movimientos, un sistema de ecuaciones lineales es construido usando (2):

$$\mathbf{A}\mathbf{x} = \mathbf{b} \quad (5)$$

donde \mathbf{A} se construye a partir de los cambios articulares $\Delta\mathbf{q}$, \mathbf{x} es un vector que contiene los elementos del Jacobiano \mathbf{J} buscado, y \mathbf{b} es construido con cambios en características $\Delta\mathbf{s}$. Con un número suficiente de movimientos se obtiene la matriz \mathbf{A} de rango completo n y los elementos de \mathbf{J} son encontrados al resolver el sistema (7).

Como lo indica la expresión (4), (a, b, c) provienen de características normalizadas, por consiguiente la calibración de las cámaras es requerida.

B. Método THRH

En este método la representación normal de las líneas es utilizada, las características (θ, ρ) son medidas directamente en la imagen y no es necesaria la calibración de las cámaras. El vector con los parámetros de las características de ambas imágenes tiene la forma:

$$\mathbf{s} = [\theta_1 \ \rho_1 \ \theta_2 \ \rho_2]^T \quad (6)$$

donde los subíndices 1 y 2 hacen referencia a la primera y a la segunda cámara. La dimensión del Jacobiano a estimarse es la misma que para el método NORM: $4M \times n$, donde M es el número de líneas, y n el número de grados de libertad. Este método también usa el método de los N últimos movimientos realizados para estimar el Jacobiano, por lo que la ecuación (7) se conforma con un número suficiente de movimientos, donde \mathbf{A} , \mathbf{x} , y \mathbf{b} representan las mismas relaciones que en el método anterior. Igualmente, los elementos de \mathbf{J} son encontrados al resolver la ecuación (7).

Como θ pertenece al intervalo $[\pi, -\pi]$ radianes, y ρ al intervalo $[0, 960]$ pixels (imagen de 768x576 pixels), se debe minimizar el sistema (7) usando el concepto de la distancia de Mahalanobis para tomar en cuenta la fuerte no similitud entre θ y ρ . La función a minimizar tiene la forma:

$$\min_{\mathbf{x}} (\mathbf{A}\mathbf{x} - \mathbf{b})^T \mathbf{W}^{-1} (\mathbf{A}\mathbf{x} - \mathbf{b}) \quad (7)$$

donde \mathbf{W} es una matriz diagonal $4M \times 4M$ definida positiva que contiene las varianzas σ^2 de θ y ρ de las líneas.

V. JACOBIANO ANALÍTICO

El Jacobiano analítico de la imagen está dado por (Corke, 1994):

$$\mathbf{J}_A = \mathbf{J}_{sr} \mathbf{W}_{cr} \mathbf{N} \mathbf{J}_R \quad (8)$$

donde \mathbf{J}_R es el Jacobiano geométrico del robot, \mathbf{N} es una matriz que contiene la cinemática directa del robot, \mathbf{W}_{cr} es la

relación entre los vectores de velocidades de la cámara y del robot, y está dado por:

$$\mathbf{W}_{cr} = \begin{bmatrix} \mathbf{R}_{cr} & [\mathbf{t}_{cr}]_x \mathbf{R}_{cr} \\ \mathbf{0}_3 & \mathbf{R}_{cr} \end{bmatrix} \quad (9)$$

siendo \mathbf{R}_{cr} y \mathbf{t}_{cr} la rotación y traslación respectivamente entre la cámara y el robot, $[\mathbf{t}_{cr}]_x$ es matriz skew-symmetric de \mathbf{t}_{cr} , y \mathbf{J}_{sr} es la matriz de interacción. Como las características de la imagen están expresadas por parámetros de líneas, \mathbf{J}_{sr} tiene la expresión [6]:

$$\mathbf{J}_{sr} = \begin{bmatrix} \lambda_\theta \cos \theta & \lambda_\theta \sin \theta & -\lambda_\rho & -\rho \cos \theta & -\rho \sin \theta & -1 \\ \lambda_\rho \cos \theta & \lambda_\rho \sin \theta & -\lambda_\theta & (1+\rho^2) \sin \theta & -(1+\rho^2) \cos \theta & 0 \end{bmatrix} \quad (10)$$

para una línea con representación normal. λ_θ y λ_ρ son obtenidos a partir de los parámetros A, B, C y d del plano 3D que contiene a la línea:

$$\begin{aligned} \lambda_\theta &= \frac{A \sin \theta - B \cos \theta}{d} \\ \lambda_\rho &= \frac{A \rho \cos \theta + B \rho \sin \theta + C}{d} \end{aligned} \quad (11)$$

La expresión (10) es para cada cámara, el Jacobiano analítico que representa a ambas cámaras es obtenido apilando los Jacobianos que pertenecen a cada cámara [8].

VI. LEY DE CONTROL

La función orientada a la tarea \mathbf{e}_k a ser regulada a cero está dada por:

$$\mathbf{e}_k = (\mathbf{s}^* - \mathbf{s}_k) \quad (12)$$

donde \mathbf{s}^* y \mathbf{s}_k son vectores de características en las posiciones deseada y en el momento k respectivamente,

Se utiliza para controlar el sistema una ley de control proporcional basada en la pseudoinversa del Jacobiano de la imagen [3] [10], en la cual el decaimiento exponencial de \mathbf{e}_k a 0 se consigue al imponer la condición $\dot{\mathbf{e}}_k = -\lambda \mathbf{e}_k$ (siendo λ un escalar positivo que regula la velocidad de convergencia). La correspondiente ley de control es:

$$\mathbf{q}_{k+1} = \mathbf{q}_k + \lambda \mathbf{J}^+ \mathbf{e}_k \quad (13)$$

siendo $\mathbf{J}^+ = (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T$ la pseudoinversa del Jacobiano que considera más características que grados de libertad.

Esta ley de control posee estabilidad asintótica local [3]. Se considera que si \mathbf{e}_k , es menor que un umbral adecuado, se entiende que el sistema ha alcanzado la posición deseada.

VII. EXPERIMENTOS

A. Equipo Experimental

El equipo experimental está compuesto por un sistema articular y un sistema de adquisición procesamiento de imágenes.

El sistema articular compuesto por un posicionador de alta precisión y su controlador modelo Newport MM3000 (Fig. 3). Posee tres grados de libertad, dos de giro y uno de desplazamiento, y su precisión teórica es de la milésima de grado y la milésima de milímetro. Sobre el último eslabón del sistema articular se ha dispuesto una pieza negra de forma triangular sobre fondo blanco, los bordes de esta pieza constituirán las líneas a detectar para controlar el sistema

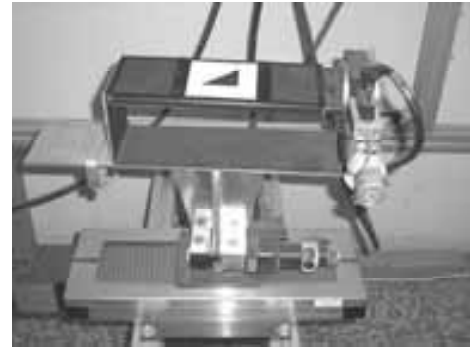


Fig. 3. Entorno de pruebas.

El sistema de visión está compuesto por dos cámaras analógicas CV-M50 y una tarjeta de adquisición de imágenes Meteor II-MC de Matrox, que permite la adquisición simultánea de las dos cámaras. Las cámaras están fijadas en el espacio de trabajo y separadas 939 mm. La detección de las características visuales se realiza con precisión subpixel, con un error menor de 0.2 píxeles. La comunicación con el controlador del sistema articular se realiza mediante un línea serie RS-232C.

B. Tarea de alineamiento

Se efectuaron tareas de alineamiento, en las cuales, a partir de una posición de inicio, las líneas que representan los bordes de la pieza triangular deben ser alineadas con las líneas de los bordes de la pieza en la posición deseada. Debido a la dependencia del Jacobiano tanto de las características visuales y de las posiciones en el espacio de las articulaciones, se ha considerado alcanzar un alto número de posiciones (hasta 50) obtenidas aleatoriamente en todo el espacio de trabajo a fin de obtener resultados más representativos.

Las características visuales deben ser alcanzables y visibles desde ambas cámaras para el espacio de trabajo considerado. Debido a que el sistema articular sólo tiene tres grados de

libertad, y a fin de asegurar coherencias, se han obtenido previamente todas las características visuales deseadas a través de la técnica de teach-by-showing (Kragic and Christensen, 2002) por el cual el sistema es movido a la posición deseada y sus correspondientes características visuales almacenadas en memoria.

Las pruebas llevadas a cabo comparan los métodos propuestos (NORM y THRH) con el método del Jacobiano analítico (ANLTC). Inicialmente, se ha considerado una y tres líneas para el control, sin ruido añadido. Posteriormente, con el fin de analizar la robustez del sistema, se ha añadido ruido gaussiano ($\sigma = 0.5$) a las características detectadas.

Para el análisis, se entiende que el sistema ha alcanzado la posición deseada si el error entre las características deseadas y actuales es menor de 0.001 para el método NORM, y de 0.05 radianes y 0.5 píxeles para el método THRH.

El error considerando una línea y tres líneas como características (con y sin ruido añadido) se muestra en la Fig. 4 y Fig. 5, y en la Fig. 6 y Fig. 7, respectivamente. Para una y tres líneas, sin ruido (Fig. 4 y Fig. 6), los tres métodos tienen un buen comportamiento además de ser muy similares. Pero con ruido añadido (Fig. 5 y Fig. 7), el método NORM se degrada en mayor medida, mientras que el método ANLTC parece ser el mejor.

Es bien conocido que el Jacobiano analítico tiene una fuerte dependencia a los errores de los modelos (ver sección 5), así que para el cálculo del Jacobiano analítico, se ha variado la distancia d del plano que contiene la línea al origen (ANLTC $d*0.7$ y ANLTC $d*1.3$). Se observa que ya sea para una o para tres líneas se produce una fuerte degradación del comportamiento del método analítico cuando la distancia d es degradada. Es evidente que aumentando el nivel de degradación, la degradación del comportamiento del Jacobiano analítico se incrementa.

VIII. CONCLUSIONES

Dos métodos para estimar el Jacobiano de la imagen utilizando líneas como características visuales han sido presentados. El primero de ellos, toma en cuenta la orientación del vector normal al plano de interpretación que contiene a la línea, el segundo método, emplea mediciones directas de las características de la línea en el plano de la imagen, utilizando la representación normal y el concepto de distancia de Mahalanobis que permite compensar las fuertes diferencias en magnitud de las características. El esquema general de estimación del Jacobiano que utilizan ambos métodos propuestos es el método de los N últimos movimientos realizados. El comportamiento de los métodos propuestos se ha contrastado con el método del Jacobiano analítico. Las pruebas realizadas indican, que en ausencia de ruido, los métodos propuestos tienen prácticamente el mismo comportamiento. En las pruebas con ruido añadido, el método analítico mostró un mejor desempeño que los métodos propuestos, además de ver que el método NORM controlando dos o tres líneas es muy sensitivo al ruido.

Hay que considerar que el método analítico trabaja bien

pero se hace necesario un laborioso y minucioso trabajo previo a la tarea de control, pues necesita la calibración de las cámaras, la transformación cámara-robot, determinación del plano que contiene a la línea, y el Jacobiano del robot. Las pruebas para el Jacobiano analítico con degradación de la transformación cámara-robot y de la distancia al plano que contiene a la línea mostraron una fuerte degradación en su funcionamiento.

Las ventajas de los dos métodos propuestos es que no necesitan un gran trabajo previo a la etapa de control, a diferencia del Jacobiano analítico que si lo necesita. Esto se considera como una ventaja y apropiados para aplicaciones en entornos desconocidos o cambiantes. En trabajos futuros, estos métodos deberán ser dotados de mayor robustez, siendo una alternativa la utilización de las propiedades de la geometría epipolar o las propiedades de las invariantes proyectivas.

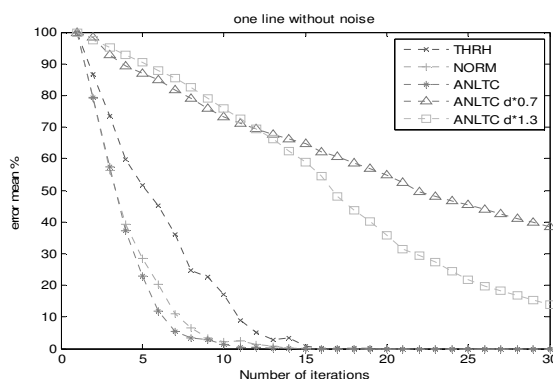


Fig. 4. Error medio en porcentaje VS número de iteraciones usando una línea, incluyendo el método analítico con degradación de la distancia d , caso sin ruido añadido.

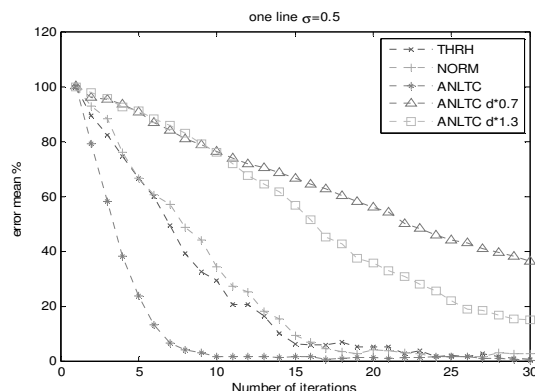


Fig. 5. Error medio en porcentaje VS número de iteraciones usando una línea, incluyendo el método analítico con degradación de la distancia d , caso con ruido añadido.

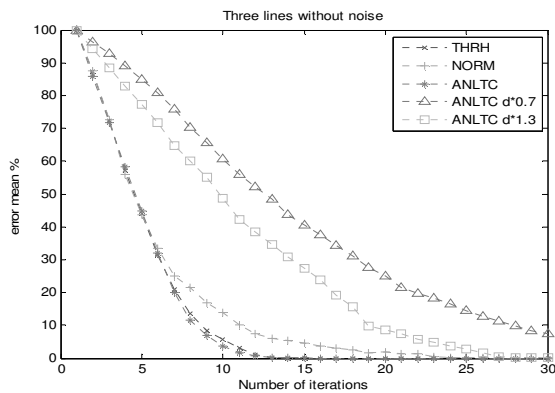


Fig. 6. Error medio en porcentaje VS número de iteraciones usando tres líneas, incluyendo el método analítico con degradación de la distancia d , caso sin ruido añadido.

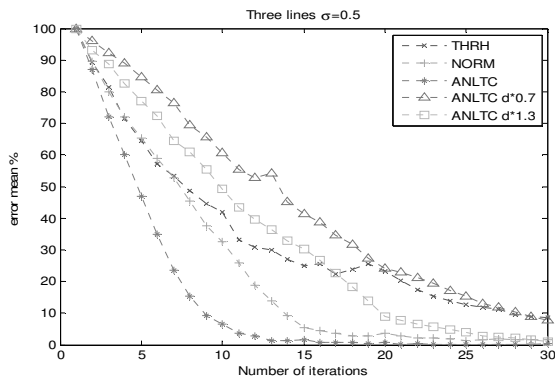


Fig. 7. Error medio en porcentaje VS número de iteraciones usando tres líneas, incluyendo el método analítico con degradación de la distancia d , caso con ruido añadido.

REFERENCES

- [1] Andreff N., Espiau B., and Horaud R. (2002). Visual servoing from lines. *International Journal of Robotics Research*. Vol. 21, No 8, pp. 679-699.
- [2] Asada, M., Tanaka, T., and Hosoda, K. (2000). Adaptive Binocular Visual Servoing for Independently Moving Target Tracking. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'00)*, 2076-2081.
- [3] Chaumette, F., and Hutchinson S. (2006). Visual Servo Control Part I: Basic Approaches. *IEEE Robotics & Automation Magazine*, vol.14, issue 1, 109-118.
- [4] Corke, P. (1994). Visual Control of Robot Manipulators – A Review. K. Hashimoto Ed., “Visual Servoing”, World Scientific, 1-32.
- [5] Deng, Z. and Jägersand, M. (2004). Evaluation of Model Independent Image-Based Visual Servoing. *Computer and Robot Vision. Proceedings. First Canadian Conference On*. 138-144.
- [6] Espiau B., and Chaumette F., and Rives P. (1992). A New Approach to Visual Servoing in Robotics. *IEEE Transactions on Robotics and Automation*, 8(3), 313-326.
- [7] Hadj-Abdelkader H., Mezouar Y., Andreff N., and Martinet P. (2006). Decoupled homography-based visual servoing with omnidirectional cameras. *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China*.
- [8] Hager G.D. (1997). A Modular System for Robust Positioning Using Feedback from Stereo Vision. *IEEE Transactions on Robotics and Automation*, Vol. 13, No 4.
- [9] Hartley, R. and Zisserman, A. (2000). *Multiple View Geometry in Computer Vision*. Cambridge.
- [10] Hutchinson, S.A., Hager, G.D. and Corke, P.I. (1996). A tutorial on visual servo control”. *IEEE Trans. Robotics and Automation*, 12-5, 651-670.
- [11] Kragic, D. and Christensen H.I. (2002). Survey on visual servoing for manipulation”. *Technical Report ISRN KTH/NA/P-02/01-Sen, CVAP25*.
- [12] Pari L., Sebastián J.M., González C., and Ángel L. (2006). Image Based Visual Servoing: A New Method for the Estimation of the Image Jacobian in Dynamic Environments. *Lecture Notes in Computer Science*, Vol 4142/2006. 850-861.
- [13] Sebastián, J.M, Pari, L., González, C. and Ángel, L. (2005). A New Method for the Estimation of the Image Jacobian for the Control of an Uncalibrated Joint System. *Lecture Notes in Computer Science*, vol 3522, 631-638.
- [14] Silveira G.F., Carvalho R.H., Rives P., Azinheira J., Bueno S., and Madrid M. (2002). Optimal visual servoed guidance of outdoor autonomous robotic airships. *Proceedings of the American Control Conference, Anchorage, AK*.
- [15] Sutanto, H., Sharma, R. and Varma, V. (1998). The role of exploratory movement in visual servoing without calibration. *Robotics and Autonomous Systems* 23. 153-169.

Aplicación de los sistemas de visión al control de movimiento de robots: Una revisión

L. Angel, Member, IEEE

Abstract—En este artículo se presentan una revisión de las diferentes estrategias empleadas para el control de movimiento de robots a través de la información suministrada por un sistema de visión. El sistema de visión esta formado por una o más cámaras y la información del entorno se adquiere en forma de imágenes. El procesamiento de imágenes permite extraer los puntos característicos del entorno, a partir de los cuales se establecen las referencias para el planificador de trayectorias del robot. Es evidente que no es un problema de fácil solución que implica la fusión de disciplinas tales como procesamiento de imágenes, cinemática, dinámica, teoría de control y computación en tiempo real. El gran volumen de datos a procesar, los tiempos consumidos por los algoritmos de procesamiento de imágenes, la dificultad de inferir información 3D a partir de una serie de imágenes 2D, etc., plantea la necesidad de disponer de algoritmos que permitan obtener soluciones en el menor tiempo posible. Desde el punto de vista de la teoría de control, los problemas anteriores se traducen en la realimentación de sistemas con retardos, la necesidad de modelado del entorno, identificación del sistema, entre otros.

Index Terms—Control Visual, Sistema de Visión, Sistema Robótico

I. INTRODUCCION

LA robótica ocupa en la actualidad un papel preponderante en el proceso de modernización e innovación de las industrias. Es conocido que ésta contribuye fundamentalmente a la calidad del producto y al mejoramiento de la productividad. Los manipuladores robóticos son una de las mejores soluciones para sustituir al hombre en procesos repetitivos, trabajos que se desarrollan en ambientes tóxicos, con factores de riesgo elevados, en tareas complejas, en tareas muy rápidas, en elaboraciones a escala micro o macroscópica, etc. Esto a motivado que una parte del ambiente científico internacional este inmerso en el desarrollo de estos mecanismos. Un aspecto relevante de este proceso, es el diseño del sistema de control encargado de guiar al robot para que ejecute la tarea encomendada con eficiencia y exactitud.

En general, un robot esta compuesto de una estructura mecánica y un sistema de control. La estructura mecánica consta de un conjunto de eslabones interconectados mediante juntas cinemáticas denominadas articulaciones. Los llamados *Robots Series*, presentan una configuración de eslabones conectados en forma secuencial, empezando por la base hasta

el extremo operativo o efector final. Cada eslabón de la cadena esta unido al anterior mediante una articulación rotacional o prismática y en todas la articulaciones hay un generador de movimiento o actuador. Un *Robot Paralelo* se define como aquella estructura en la cual, el efector final esta unido a la base por más de una cadena cinemática cerrada independiente. Esta configuración, confiere a los robots paralelos ventajas con respecto a los robots series en términos de rigidez, velocidad, precisión, inercia en movimiento y relación carga/peso. La mayor desventaja de este tipo de manipuladores, es su limitado rango de movimiento.

En cuanto al sistema de control, esta compuesto por los actuadores, los sensores, el controlador, el planificador de trayectorias, y el sistema de supervisión y decisión. Los actuadores son los encargados de accionar cada una de las articulaciones con el fin de posicionar y orientar el efector final. Estos pueden ser de tipo eléctrico, neumático o hidráulico. Los sensores por su parte, son los encargados de medir las variables de interés para el control del robot. Los sensores pueden ser de dos clases: los que proveen el reconocimiento del estado mecánico del manipulador (por ejemplo: sensores de posición, velocidad, fuerza), y los que permiten una realimentación del entorno del manipulador (por ejemplo: cámaras, sensores de ultrasonidos, etc.). El controlador recibe las consignas de referencia de las variables asociadas a la trayectoria del extremo operativo, generadas por el sistema de planificación de trayectorias. De acuerdo a la información entregada por los sensores, se generan las acciones de control adecuadas para cada actuador según el algoritmo de control empleado.

II. CONTROL VISUAL: CONCEPTOS BÁSICOS

La mayoría de los sistemas robotizados operan en fábricas, donde el espacio de trabajo ha sido ideado para adecuarse al robot. Sin embargo los robots han tenido menos impacto en aplicaciones dentro de entornos dinámicos, donde el área de trabajo y la ubicación de los objetos no pueden controlarse exactamente. Esta limitación se debe implícitamente a la falta de capacidad sensorial en los sistemas robóticos comerciales. La integración sensorial es fundamental para incrementar la versatilidad y el dominio de aplicación de los robots.

Actualmente, los sensores visuales son cada vez mas frecuentes en las aplicaciones robóticas. Su principal ventaja es que permiten una descripción muy completa del entorno de forma no intrusiva. Se puede decir, que facilitan la integración de dispositivos robóticos en entornos menos estructurados y

L. Angel is with the Universidad Pontificia Bolivariana, Bucaramanga, Colombia, email: lasilva@upbbga.edu.co.

dinámicos, que los que se encuentran en las aplicaciones clásicas de automatización. La expresión control visual se aplica a sistemas guiados visualmente, los cuales hacen uso de una o varias cámaras para obtener información del entorno en forma de imágenes, que es empleada para controlar el efector final del robot durante la realización de una tarea. Según Weiss [1], los sistemas de control visual pueden ser clasificados a partir de tres criterios: la configuración física del sistema de visión, la utilización de las características extraídas de la imagen y la arquitectura del robot.

La configuración física del sistema de visión, define la relación cinemática del sistema de visión con respecto al robot y al entorno. El sistema de visión puede estar formado por uno o varios sensores visuales y puede tratarse de cámaras estáticas y/o móviles. La disposición del sistema de visión permite distinguir dos tendencias diferenciadas [2]: cámaras en un lugar fijo de la escena (*eye-to-hand*) y cámaras en el extremo del robot (*eye-in-hand*).

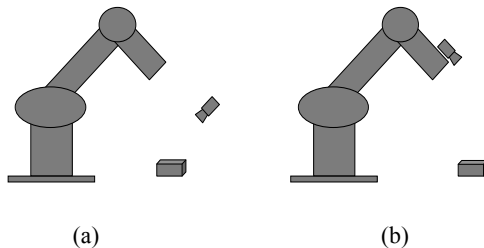


Fig. 1. Disposición del sistema de visión en control visual: (a) cámaras en un lugar fijo de la escena, (b) cámaras en el extremo del robot.

En el caso de cámaras en un lugar fijo de la escena (Fig. 1(a)), el sistema de visión se encuentra dispuesto en una ubicación fija y observa la escena en la que deben estar permanentemente visibles tanto el robot (típicamente, el efector final) como el objeto de interés. No existe una relación mecánica entre el sistema de visión y el robot, pero es conocida la relación entre la cámara y el sistema de referencia asociado a la base del robot. Para la configuración de cámaras en el extremo del robot (Fig. 1(b)), el sistema de visión es móvil y se encuentra solidario al efector final. La relación entre la localización de la cámara y la del efector final del robot es conocida y constante. La atención se centra en la observación del objeto, mientras que la disposición del robot queda implícita en la información extraída de las imágenes. El empleo de cámaras estáticas, presenta como principal desventaja la eventual *oclusión* que puede producirse cuando el robot se interpone entre la cámara y el objeto.

Desde el punto de vista del procesamiento de imágenes, las principales estrategias para realizar el control visual son: control basado en posición (control visual 3D), control basado en imagen (control visual 2D), o una mezcla de los dos anteriores (control visual 2½D).

En el control visual basado en posición (Fig. 2), las características extraídas de las imágenes, s , son usadas junto al modelo geométrico del objeto y el conocimiento del modelo de la cámara, para estimar explícitamente la localización del

objeto respecto al sistema de visión, x . La señal de error se genera en el espacio 3D. Como principales ventajas de este método destacan: la tarea se especifica en términos de localización cartesiana similares a los empleados en robótica, la trayectoria de la cámara es una línea recta, ausencia de singularidades en el espacio de la tarea y mínimos locales prácticamente inexistentes. Como desventajas destacan, la excesiva dependencia de los errores de calibración debidos a la calibración de la cámara o al modelo geométrico impreciso del objeto, y el uso de información adicional para reconstruir las coordenadas cartesianas de los puntos en el espacio.

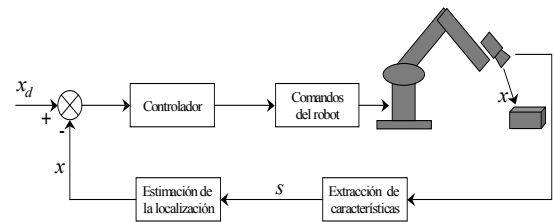


Fig. 2. Control visual basado en posición.

En los sistemas de control basados en imagen (Fig. 3), el control se realiza directamente a partir de las características extraídas por el sistema de visión, que representan la proyección del objetivo a alcanzar en el plano de la imagen. De esta forma, la señal de error se define directamente en términos de las propias características de la imagen. En esta estrategia de control, el concepto clave es el de *matriz de interacción* o *matriz jacobiana de la imagen*. Esta matriz establece la relación mediante la cual, un cambio en la posición y orientación del objeto respecto a la cámara induce un cambio en las características observadas en la imagen. Esta matriz puede estimarse *off-line* o estimarse en cada iteración de la ley de control. La principal ventaja de esta estrategia de control es la precisión obtenida independiente del proceso de calibración y del conocimiento del modelo del objeto. Entre los principales inconvenientes están: trayectoria de la cámara no controlable, no se puede asegurar la estabilidad global y la aparición de singularidades en el espacio de características, como oclusiones parciales.

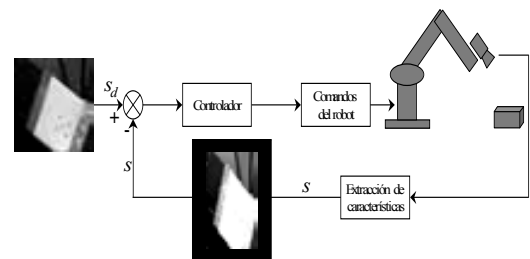


Fig. 3. Control visual basado en imagen.

El control visual 2½D propuesto por Malis et al. [3] está a

medio camino entre el control visual basado en posición y el control visual basado en imagen, explotando las ventajas de ambos. Además, permite evitar sus desventajas respectivas, ya que contrariamente al control visual basado en posición, no necesita ningún modelo geométrico del objeto y en comparación con el control visual basado en imagen, asegura la convergencia de la ley de control en todo el espacio de trabajo.

A partir de la arquitectura del robot [4], los diversos esquemas de control visual se clasifican en: control visual indirecto y control visual directo. En el esquema de control visual indirecto están presentes además del lazo de realimentación visual, los bucles internos de control de los servomotores del robot. Este esquema de control comúnmente se denomina “mirar y mover” (*look-then-move*) y puede ser estático o dinámico.

En el esquema “*mirar y mover estático*”, Fig. 4, los algoritmos de visión y el movimiento del robot operan en forma secuencial. En primer lugar, la cámara captura una imagen, extrayéndose la información necesaria para dar el próximo comando de movimiento al robot. Hecho esto, se envía la orden de movimiento al robot y se espera a que haya finalizado dicho movimiento antes de tomar una nueva imagen. El robot ejecuta la tarea suponiendo que el entorno no se modificó. La dinámica de los bucles internos de control de los servomotores del robot, en tanto sea estable, no tiene ninguna influencia sobre el bucle de control visual.

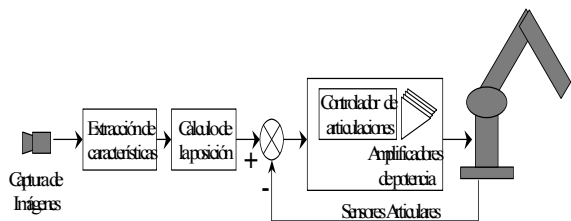


Fig. 4. Esquema de control visual “*mirar y mover estático*”

En la variante “*mirar y mover dinámico*”, Fig. 5, el robot puede encontrarse aun en movimiento mientras la siguiente imagen esta siendo tomada. En este caso, se tiene en cuenta la dinámica de los bucles internos de control de los servomotores del robot, lo que lleva a implementaciones de control bifrecuencial, en el cual, el lazo de control visual opera a una frecuencia más baja, proporcionando valores de referencia al bucle de control interno, de mayor frecuencia.

En el esquema de control visual directo, Fig. 6, comúnmente llamado control “*servo-visual*” (*visual servoing*), desaparece el lazo de control a nivel articular. El bucle de realimentación visual controla directamente los servomotores del robot, lo que genera un fuerte nivel de acoplamiento dinámico en el lazo de control visuo-motor.

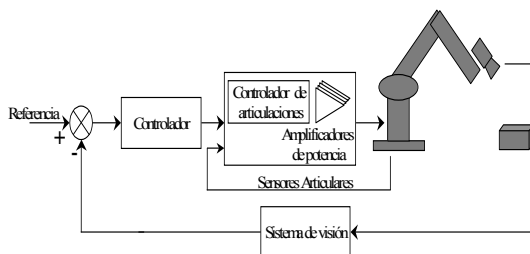


Fig. 5. Esquema de control visual “*mirar y mover dinámico*”

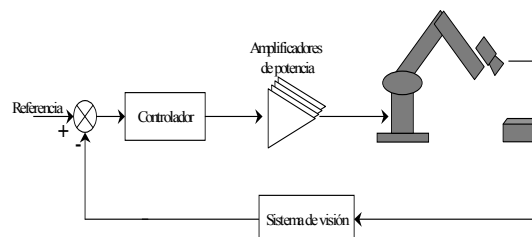


Fig. 6. Esquema de control visual directo

III. CONTROL VISUAL: REVISIÓN Y FUTUROS DESAFÍOS

Las primeras aplicaciones de control visual datan de la década de los 70 y consisten en sistemas de tipo “ver y mover” estático. Una de estas primeras aplicaciones de guiado de robots por visión fue la desarrollada por Shirai y Inoue [5], quienes describen como puede utilizarse la realimentación visual para corregir la posición de un robot y aumentar la precisión. Esta primera aplicación consistía en utilizar realimentación visual para el guiado de un robot de forma que realice el agarre de un prisma y su posterior inserción en una caja. Durante esta década se desarrollaron las primeras aplicaciones de control visual para el agarre de objetos en movimiento por cintas transportadoras destacando asimismo los trabajos de Hill y Park [6] que fueron quienes presentaron en 1979 el concepto de “visual servoing”.

En 1987, Sanderson y Weiss [7] establecen una clasificación de los sistemas de control visual, describiendo asimismo el primer sistema de control visual basado en imagen. Desde la aparición de estos primeros sistemas de control visual a mediados de los 80, su evolución ha sido lenta, sin embargo en los últimos años han aparecido crecientemente nuevos desarrollos debidos principalmente a las capacidades de procesamiento de los nuevos ordenadores personales que permiten procesar una escena a una frecuencia cada vez mayor.

Ya a finales de los 80 comienza a aplicarse el control visual a muy distintos ámbitos [8] como puede ser el agrícola, en el que se encuentran aplicaciones para la recogida de frutas empleando realimentación visual [9], o la teleoperación de robots. Dentro de este último ámbito el control visual permite al operador humano especificar la tarea en términos de

características en la imagen [10] [11].

Otro ámbito de interés en control visual es el de la vigilancia. En muchas ocasiones el sistema de visión se localiza sobre una estructura móvil de forma que es capaz de mantener en el campo de visión un determinado objeto que se encuentra bajo vigilancia. Como se comentó en el apartado de control visual basado en imagen, en este tipo de aplicaciones es crítica la estimación del movimiento del objeto seguido.

Durante los 90 y con el aumento de las capacidades de los sistemas de visión artificial aparecen un gran número de aplicaciones. Dentro de este nuevo conjunto de aplicaciones cabe mencionar algunas como el guiado de vehículos [12] en las que se requiere una gran capacidad de procesamiento y una alta frecuencia en la realimentación visual. Otro ámbito en el que el control visual presenta una importancia creciente en la actualidad es el del guiado de robots aéreos. Aplicaciones dentro de este ámbito no han sido explotados en gran medida, sin embargo, ya empiezan a surgir vehículos aéreos, como helicópteros [13], dotados de un sistema de visión que realimenta al controlador del vehículo de forma que permite realizar su guiado a partir de características del entorno, evitando obstáculos e incluso permitiendo realizar un aterrizaje automático. El desarrollo de sistemas de control visual en estos vehículos permite realizar aplicaciones como vigilancia, seguimiento, inspección, trabajo en lugares peligrosos, etc.

Con la miniaturización y el aumento de precisión de los componentes de los robots, van aumentando las posibilidades de la micromanipulación empleando realimentación visual procedente, por ejemplo, de microscopios. Se vienen aplicando en la actualidad sistemas de control visual con el objetivo de mantener en la imagen a lo largo del tiempo ciertos microorganismos con el propósito de su posterior manipulación [14]. Existen aplicaciones a robots submarinos [15]. En este caso se dispone de vehículos submarinos guiados por visión o sistemas de manipulación submarina cuya tarea es guiada mediante realimentación visual.

Un ámbito de aplicación con mayor impacto social son las aplicaciones médicas [16]. La alta precisión alcanzada con estos sistemas ha propiciado la aparición de sistemas para la realización de telecirugía así como robots que realizan tareas quirúrgicas de forma autónoma. A pesar de que estos sistemas aumentan el coste por intervención, la alta precisión junto con el pequeño espacio de trabajo requerido en, por ejemplo, cirugía cerebral, hace que estos sistemas se apliquen con éxito.

Finalmente, es importante resaltar los nuevos desafíos del control visual mediante aplicaciones al control de movimiento de robots paralelos. En este campo se destaca el Sistema RoboTennis, una plataforma abierta construida con el fin de desarrollar nuevas estrategias de control visual que permitan la integración de sistemas robóticos paralelos y sistemas de visión en la realización de tareas con altos requisitos temporales [17]. Actualmente, este sistema realiza el seguimiento 3D de una pelota que se mueve a velocidades hasta de 1.5 m/s [18].

Un factor a tener en cuenta en aplicaciones de control visual, es la dinámica del entorno. Las aplicaciones en entornos estructurados con presencia de objetos cuya posición y orientación es perfectamente conocida, es un problema suficientemente estudiado en la actualidad. Sin embargo, aplicaciones en las que la variabilidad del entorno añade un alto grado de incertidumbre al mismo, presenta numerosas dificultades aun no suficientemente resueltas. La realización de tareas robóticas en este tipo de entornos, requieren de estructuras robóticas de bajo peso y alta rigidez, actuadores que permitan altas aceleraciones y altas velocidades, sensores con procesamiento rápido y esquemas de control sofisticados que tomen en cuenta la dinámica no lineal del sistema.

REFERENCES

- [1] Weiss, L.E. Dynamic Visual Servo Control of Robots: an Adaptive Image-Based Approach. PhD thesis, Carnegie-Mellon University, 1980.
- [2] Hutchinson, S., Hager, G.D., Corke, P.I. A tutorial on visual servo control, *IEEE Transactions Robotics and Automation*, vol. 12, No. 5, 651-670, 1996.
- [3] Malis, E., Chaumette, F. and Boudet, S. 2-1/2-D visual servoing, *IEEE Transactions Robotics and Automation* 15 (2), 238-250, 1998.
- [4] Sanderson, N. Dynamic sensor-based control of robots with visual feedback, *IEEE Journal of Robotics and Automation*, vol.3, No. 9, 1987, 404-417, 1987.
- [5] Shirai, Y. and Inoue, H. Guiding a robot by visual feedback assembling tasks. *Pattern Recognition*, vol. 5, 99-108, 1973.
- [6] Hill, J. and Park, W.T. Real time control of a robot with a mobile camera., *9th ISIR*. Washington D.C, 233-246, 1979.
- [7] Weiss, L. E., Sanderson A. C. and Newman, C. P. Dynamic sensor-based control of robots with visual feedback, *IEEE Journal Robotics and Automation*, vol. 5, 404-417, 1987.
- [8] Corke, P. I. "Visual control of robot manipulators – a review". Visual servoing. K. Hashimoto, Ed. Singapore: World Scientific, 1-31. 1993.
- [9] Harrell, R. C.; Slaughter, D. C.; Adsit, P. D. A fruit-tracking system for robotic harvesting, *Machine Vision and Applications*, 69-80. 1989.
- [10] Vikramaditya, B.; Nelson, B. J. Visually Guided Microassembly Using Optical Microscopes and Active Vision Techniques, *IEEE Int. Conference on Robotics and Automation*, 3172-3177, 1997.
- [11] Papanikolopoulos, N. P.; Khosla, P. K. Shared and traded telerobotic visual control, *IEEE International Conference Robotics and Automation*, 878-885, 1992.
- [12] Dickmanns, E. D.; Graefe, V. Applications of dynamic monocular machine vision, *Machine Vision and Applications*, vol. 1, 241-261, 1988.
- [13] Chriette, A.; Hamel, T.; Mahony, R. Visual servoing for a scale model autonomous helicopter, *IEEE International Conference on Robotics and Automation*, Seoul, Korea, 21-26, 2001.
- [14] Oku, H.; Ishii, I.; Ishikawa, M. Tracking a Protozoon Using High-Speed Visual Feedback, *1st Annual International. IEEE-EMBS Special Topic Conference on Microtechnologies in Medicine & Biology*, Lyon, France, 156-159, 2000.
- [15] Lots, J. F.; Lane, D. M.; Trucco, E.; Chaumette, F. A 2-D Visual Servoing for Underwater Vehicle Station Keeping, *IEEE Conference on Robotics and Automation*, Seoul, Korea, 2767-2772, 2001.
- [16] Wei, G. Q., Arbter, K., Hirzinger G. Real-Time Visual Servoing for Laparoscopic Surgery, *IEEE Engineering in Medicine and Biology*, vol. 16, No. 1, 40-45. 1997.
- [17] Angel, L., Sebastian, J.M., Saltaren, R., Aracil, R., Sanpedro, J. RoboTennis: Optimal Design of a Parallel Robot with High Performance, In Proc. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'05)*, Edmonton, Canada, August, 2005.
- [18] Traslosheros, A., Sebastián, J.M., Angel, L., Roberti, F., Carelli, R. Visual Servoing using a Parallel Robot: Preliminary Results, In Proc. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM'07)*, Zurich, Switzerland, September, 2007.

Simulaciones para el aprendizaje de comunicaciones analógicas y digitales

Pedro Vizcaya G., miembro IEEE, Alba Lucía Meneses

Resumen— El aprendizaje de los conceptos modernos de las comunicaciones digitales y sus técnicas particulares es cada vez más difícil por el gran volumen de teorías (de señales y de comunicaciones) y de desarrollos en todos los componentes de un sistema de comunicación actual. Prueba de esto es el aumento en el número de temas y el volumen de los textos empleados para su enseñanza. Por otra parte la reforma del sistema de créditos actual propende por una menor presencialidad y una mayor responsabilidad individual en el proceso de aprendizaje. Finalmente, en la enseñanza de la ingeniería es muy importante mantener en enlace entre la teoría y la práctica, lo cual es cada vez más difícil por los altos costos, rápida obsolescencia y falta de flexibilidad de los equipos de comunicaciones comerciales. En el presente artículo se presenta una experiencia docente en donde se rescata el uso de simulaciones para estudiar sistemas de comunicación digitales a través de prácticas dirigidas que le permiten al estudiante experimentar con sistemas de complejidad variada a partir de bloques básicos y de instrumentación virtual. Estas prácticas de simulación le facilitan al estudiante la integración conceptos y técnicas simples en sistemas complejos y la evaluación de su comportamiento ante los disturbios normales que se presentan en los sistemas reales, algo que no es posible hacer en forma eficiente a partir de la teoría únicamente. Estas prácticas van acompañadas de una guía que inicia rápidamente al estudiante en la operación de la misma, al mismo tiempo que orienta sobre los puntos importantes ilustrados, sin más ayuda que la recibida en la clase y su texto de estudio, lo cual hace de la práctica un material valioso para fomentar el autoaprendizaje, sin demandar un esfuerzo ni un tiempo desproporcionados. Al presente se han desarrollado 18 prácticas que cubren los principales temas de un curso de comunicaciones analógicas y digitales típico.

Palabras clave— Enseñanza-aprendizaje de Sistemas de Comunicación, Simulación de sistemas de comunicación.

I. INTRODUCCIÓN

Los temas considerados como básicos en un curso de fundamentos de comunicaciones han aumentado considerablemente en cantidad y calidad en los últimos 30 años, juzgando por la evolución de los textos más empleados en pregrado para este tema ([1][2], [3], y [4]), los cuales

Manuscrito enviado el 11 de Julio de 2008. Este trabajo ha sido apoyado en parte por la Vicerrectoría Académica de la P. Universidad Javeriana a través de la Convocatoria: Experiencias para el mejoramiento de la docencia, 2007, y por el Departamento de Electrónica.

P. Vizcaya es professor titular de la Facultad de Ingeniería de la P. Universidad Javeriana (pvizcaya@javeriana.edu.co) del área de Comunicaciones y Análisis de Señales.

A. L. Meneses es docente de las universidades Católica y El Bosque del área de Procesos Cognitivos y Metodología de la Investigación.

cuentan ya con 4 o más ediciones. Típicamente, en un currículo norteamericano de pregrado se tiene un único curso de introducción a comunicaciones que debe cubrir tanto los temas de comunicaciones análogas como digitales, usualmente sin contar con el requisito de un curso de probabilidad y a veces simultaneo con un curso introductorio de señales y sistemas. Aún habiendo cubierto estos prerrequisitos, el contenido sugerido para un curso suele dejar de lado algunos conceptos fundamentales de teoría de decisión o la teoría de señales, como lo insinúa Carlson en el prefacio de su texto [1], o la introducción a temas de actualidad, como lo señala Haykin en el prefacio de su texto [2] En cualquier caso, y particularmente cuando el plan de estudios Ingeniería Electrónica considera solo un curso de un semestre para el tema, para el instructor se presenta el dilema de cubrir pocos temas de actualidad pero trabajar con más profundidad los conceptos básicos (con la esperanza de que el estudiante podrá por si mismo complementar su formación) o cubrir más temas de actualidad con menor rigor formal (con la expectativa de que el estudiante interesado en el área complete su formación en las asignaturas de énfasis o postgrado del área).

El presente artículo ofrece una experiencia de enseñanza que emplea simulaciones en Simulink como apoyo para el aprendizaje/enseñanza de temas específicos de comunicaciones, que pueden facilitar la presentación eficiente de estos temas. Dos de los textos mencionados ([2] y [3]) incluyen problemas y ejemplos trabajados con Matlab, pero no aprovechan las cajas de herramientas (*toolboxes*) de comunicaciones y señales de Simulink, el simulador interactivo orientado a bloques de Matlab. Más recientemente, Proakis y otros [5] presentan un texto de sistemas de comunicación con un uso profuso de Matlab básico (primera edición) al cual se le adicionó un capítulo con Simulink en la segunda edición, reconociendo así la utilidad de este programa para presentar en forma gráfica e interactiva los conceptos de comunicaciones. Adicionalmente se puede mencionar la publicación de Jeruchim y otros [6], la cual estudia el tema de la simulación y el modelamiento de sistemas de comunicación como tal, aunque no desde la perspectiva de la enseñanza de las comunicaciones.

Instructores en esta área de las comunicaciones hemos empleado herramientas como Matlab y Simulink para apoyar la labor docente, principalmente a través de la asignación de ejercicios y problemas más o menos abiertos para solucionarlos empleando este lenguaje de programación. Estos

ejercicios normalmente no han sido publicados, citando como ejemplo a Edwards [7] y Kang y otros [8]. En mi práctica con esta aproximación he encontrado que, dada la existencia de múltiples soluciones válidas para estos problemas y consistentes internamente usando Matlab/Simulink y, por supuesto, la mayor cantidad de posibilidades de error, la evaluación detallada de estos trabajos se torna impráctica e ineficiente: en primer lugar, el estudiante debe enfrentar a la vez dificultades desde al menos dos frentes (la teoría de las comunicaciones y el manejo de la herramienta de simulación), lo cual aumenta las posibilidades de error y el tiempo empleado, mientras que el instructor debe tener el conocimiento necesario para ser capaz de corregir los errores consistentemente con la teoría de las comunicaciones y con el camino para resolver el problema particularmente adoptado por cada estudiante. Este proceso fácilmente falla por falta del recurso más limitado, el tiempo, impidiendo que se cumpla apropiadamente el ciclo de aprendizaje.

La metodología que se presenta en este trabajo tiene en cuenta las anteriores limitaciones. En cuanto al alcance de las prácticas es comparable con las presentadas por Krishnaswamy y Ramu [9], pero se ha puesto más atención en limitar el tiempo que su desarrollo demanda de los estudiantes, sacrificando los detalles particulares de la simulación y dándole prioridad a los conceptos de comunicaciones. En lo que sigue se presenta la metodología desarrollada, ilustrándola con un ejemplo, las características generales de las prácticas desarrolladas, se presentan algunos resultados y conclusiones.

II. PRESENTACIÓN DE LA METODOLOGÍA

La metodología propuesta para el desarrollo de estas prácticas de simulación está directamente encadenada con el desarrollo de la clase e incluye los siguientes pasos:

- 1) Preparación de la teoría por los estudiantes (autoestudio).
- 2) Evaluación de control de lectura de la teoría.
- 3) Revisión de la teoría en clase y solución de dudas.
- 4) Presentación general de la práctica de simulación.
- 5) Ejecución de la simulación por parte de los estudiantes (estudio independiente).
- 6) Solución de dudas de la simulación en clase.
- 7) Evaluación de la simulación.

Esta metodología se ilustra con el siguiente ejemplo. Para esta simulación el estudiante ha recibido el archivo LloydMaxVoz.mdl, un archivo inicial para entrenar el cuantificador escalar denominado Lloyd-Max-Voz.sqd y un archivo con una muestra de señal de voz a 8kHz y 32 bits flotante, con duración de un minuto. Esta muestra es un segmento de una grabación realizada a 44.1kHz, 16kb/s, la cual fue filtrada y diezmada a la tasa final, con lo cual simula perfectamente muestras tomadas de una señal continua. Al abrir la simulación con Simulink (ver Figura 1), el estudiante

accede a un archivo de instrucciones con formato html que lo guiará a través de la misma. Su contenido se muestra en la **¡Error! No se encuentra el origen de la referencia.**

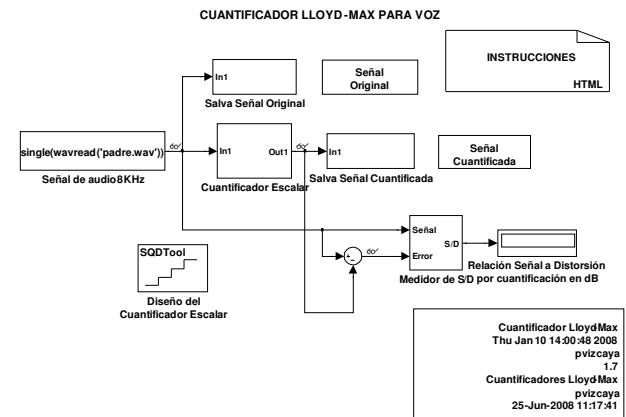


Figura 1. Simulación de cuantificación escalar.

En esta guía en los pasos 2 y 3 se indica el procedimiento necesario para iniciar el bloque Cuantificador Escalar usando el bloque de Diseño de Cuantificador Escalar. En el paso 4 se indica como correr la simulación y evaluar la señal cuantificada tanto cuantitativamente (relación señal a distorsión) como cualitativamente (escuchando la señal por los dispositivos de multimedia del computador y viendo la señal generada con los visores conectados en los puntos de entrada y salida correspondientes, como se ilustra en la Figura 3).

En el paso 5 se le indica al estudiante como evaluar la relación S/D para diferente número de pasos de cuantificación. En el paso 6 se le indica al estudiante cómo entrenar el cuantificador para distribuciones uniforme y gaussiana. Finalmente en el paso 7 se le pide al estudiante entrenar el cuantificador escalar con la señal original para obtener el cuantificador óptimo para ella. Se le pide además consignar los resultados obtenidos en la tabla adjunta. A lo largo de estas instrucciones se le solicita al estudiante estudiar los diferentes parámetros de la simulación y se le hacen preguntas directas sobre los resultados obtenidos.

Se aspira a que el estudiante siga los pasos de esta guía por si mismo (Paso 5) de la metodología) y así esté en capacidad de participar activamente con preguntas en la sesión de revisión de la simulación (Paso 6 de la simulación). Finalmente esta simulación se evalúa individualmente en la sesión de clase posterior a anterior, a través de un cuestionario de 6 preguntas de selección múltiple que están enfocadas tanto a aspectos operativos de la simulación como a las conclusiones sobre las preguntas que se hacen en la guía.

Cuantificación Lloyd/Max para Señales de Voz

Objetivo

Entrenar, evaluar y comparar un cuantificador Lloyd-Max para señales de voz con relación al cuantificador uniforme.

Específicos

1. Entrenar y evaluar el cuantificador Lloyd-Max para distribución uniforme.
2. Entrenar y evaluar el cuantificador Lloyd-Max para distribución de voz.

Desarrollo:

1. Abra la simulación LloydMaxVoz.
2. Entrenamiento del cuantificador Lloyd-Max con distribución uniforme.
 - Haga doble clic sobre el bloque de Diseño del Cuantificador Escalar. Esto abre la sesión de diseño por defecto.
 - Por el menú *File* abra la sesión predefinida denominada Lloyd-Max-Voz.sqd, la cual está en un archivo independiente, suministrado con esta simulación.
 - Con la ayuda de Simulink, asegúrese de entender el uso de los diferentes parámetros y opciones del entrenador.
 - Entrene el cuantificador haciendo clic en la opción *Design and Plot*. Haciendo clic sobre la curva de error lea su valor con precisión. Puede salvar las variables del cuantificador diseñado en el área de trabajo principal de MATLAB en cualquier momento, haciendo clic en la opción *Export Outputs* y habilitando la sobre-escritura de las variables existentes.
 - Exporte los parámetros del cuantificador diseñado al área de trabajo empleando la opción *Export Outputs*.
3. Identifique cómo se usan las variables generadas por el bloque de Diseño del cuantificador Lloyd-Max. Identifique igualmente todas las entradas y salidas del bloque Cuantificador Escalar. Verifique también cómo se mide la relación señal a distorsión (S/D).
4. Corra la simulación por 10 segundos y tome note de la relación S/D. Observe las señales Original, Error y Cuantificada en el visor Scope. Escuche la señal original y la cuantificada haciendo doble-clic en el bloque correspondiente.
5. Repita los pasos 2 al 4 para los siguientes valores del número de niveles del cuantificador: 16, 32, 64, 128 y 256. ¿Para que número de niveles el ruido de cuantificación no es perceptible? ¿Cuántos dB de diferencia en la relación S/D hay en cada paso?
6. Ahora repita los pasos 2 al 5 empleando como conjunto de entrenamiento una secuencia con distribución gaussiana. Para ello escriba $0.25 * \text{randn}(10000,1)$ en lugar de $\text{rand}(10000,1) * 2 - 1$ en la opción Training Set de la ventana SQ Design Tool. ¿Qué diferencia en la relación S/D encuentra con relación al cuantificador uniforme para cada uno de los casos de número de niveles de cuantificación analizados?
7. Finalmente repita los pasos 2 al 5 empleando como conjunto de entrenamiento la señal original, la cual se encuentra salvada en la variable *y_org*. Para ello escriba *y_org* en lugar de $\text{rand}(10000,1) * 2 - 1$ en la opción Training Set de la ventana SQ Design Tool. ¿Qué diferencia en la relación S/D encuentra con relación al cuantificador uniforme para cada uno de los casos de número de niveles de cuantificación analizados?
8. Llene la siguiente

Tabla 1. Relación Señal a Distorsión (dB)

L→	16	32	64	128	256
Uniforme					
Gausiana					
Lloyd-Max					

Figura 2. Guía de la práctica de simulación.

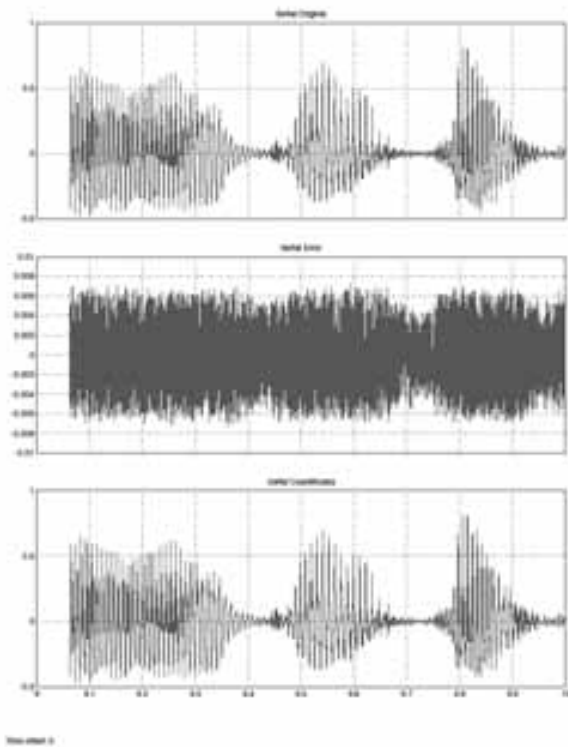


Figura 3. Visualización de las Señales del Cuantificador Escalar

III. EVALUACIÓN DE LAS SIMULACIONES

Se diseñó y aplicó un cuestionario tipo Likert, de evaluación sobre las simulaciones desarrolladas en los Cursos de Sistemas de Comunicación I y II, durante el primer periodo académico de 2008. Primero, se realizó un grupo focal conformado por 4 estudiantes, dos de cada curso (uno con un rendimiento académico alto y el otro con un rendimiento académico bajo) y una estudiante de Maestría que apoyaba la implementación de las simulaciones en dichas asignaturas, con el propósito de determinar los aspectos a mejorar de la didáctica de las simulaciones y como primer insumo para generar los ítems del cuestionario de tipo cerrado.

Con base en el análisis realizado a los datos obtenidos en el grupo focal se diseñó un cuestionario de tipo cerrado con 48 preguntas que evaluaba entre otros aspectos nivel de motivación y de actitudes hacia las simulaciones, pertinencia del contenido de las guías, requerimientos para desarrollar correctamente las simulaciones, pertinencia de las evaluaciones de las simulaciones, sugerencias para mejorar las simulaciones como contexto de aprendizaje de los contenidos teóricos de las asignaturas, promedio de horas para el desarrollo de las simulaciones y promedio de horas para la preparación de las asignaturas.

A continuación se presenta una breve descripción de las características de los estudiantes que cursaron cada una de las

asignaturas de Sistemas de Comunicación I y II durante el primer período académico de 2008.

El curso I estuvo conformado por 10 hombres y una mujer con un promedio de edad de 22.6 años y una desviación estándar de 1.69, uno de los estudiantes repetía la asignatura por segunda vez. El curso II estuvo conformado por 17 hombres y cuatro mujeres con un promedio de edad de 22.6 años y una desviación estándar de 1.4. El 52% de los estudiantes eran repitentes, de los cuales cinco tomaban la asignatura por tercera vez (23%), cuatro por segunda vez (19%), uno por cuarta (5%) y otro por quinta vez (5%).

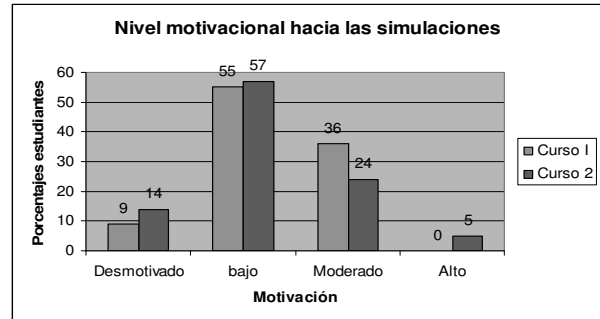


Figura 4. Motivación hacia las Simulaciones.

En seguida se presentan los resultados de la evaluación realizada, así como una breve descripción del instrumento de medida.

El nivel motivacional hacia las simulaciones se midió con una prueba tipo Likert de nueve preguntas (1 = desmotivación a 5 = alta motivación), con un nivel de consistencia interna α de Cronbach de 0.74. Los resultados muestran que la mayoría de los estudiantes de ambos cursos reportan presentar bajo y moderado nivel motivacional (ver Figura 4).

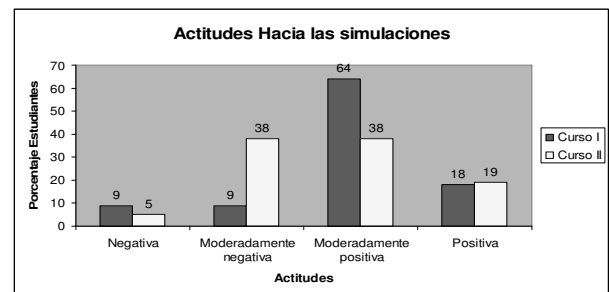


Figura 5. Actitudes hacia las Simulaciones.

El nivel de actitudes hacia las simulaciones se midió con una escala Likert de 11 preguntas (1= actitud negativa a 5=actitud positiva), el nivel de consistencia interna medido por el coeficiente de α de Cronbach fue de 0.85 El reporte del Curso I evidencia una mejor actitud hacia las simulaciones que el grupo II, el primero se halla en un nivel de actitud entre moderadamente positivo (64%) y positiva (18%) y el curso II presenta entre un 38% de actitud moderadamente positiva y un 19% positiva (ver Figura 5).

Se evaluó la pertinencia del contenido de las guías de las simulaciones a través de dos preguntas y las respuestas fueron:

En el curso I, el 36% considera medianamente pertinente o pertinentes los contenidos para desarrollar las simulaciones y lograr los aprendizajes propuestos. En el grupo II, el 66% los considera medianamente pertinentes o pertinentes. Cuando se evaluó si el contenido de las guías de las simulaciones presentaban referencias conceptuales de los temas a desarrollar; así como pasos claros y completos: El curso I (64%) y el curso II (52%) consideran que los presenta entre moderadamente a suficientes.

Se evaluaron los requerimientos que consideran necesarios los estudiantes para el desarrollo satisfactorio de las simulaciones con un cuestionario de 9 preguntas tipo Likert. Los estudiantes del curso I (85%) y del curso II (86%) consideran necesarios para el desarrollo satisfactorio de las simulaciones: la presentación teórica del profesor, asistir a clase y tomar apuntes, comprender el libro de texto, poseer conocimientos básicos de MATLAB, tener adecuado manejo de SIMULINK, poder resolver dudas en la clase propuesta para ello y tener la retroalimentación del profesor para comprender los errores y hacer los ajustes a que haya lugar, luego de haber realizado la simulación.

La pertinencia del contenido de las evaluaciones se evaluó a través de tres preguntas tipo Likert. La mayoría de los estudiantes del curso I (71%) y del curso II (76%) están de acuerdo en el que haber desarrollado la simulación les permite responder la evaluación correctamente y en relación con los objetivos de aprendizaje de las simulaciones y que las evaluaciones permiten al profesor saber si el estudiante las ha realizado.

Se evaluó con un cuestionario de nueve preguntas, aspectos a mejorar de las simulaciones. La mayoría de los estudiantes (85%) de los cursos I y II están de acuerdo que para mejorar la calidad de las simulaciones las preguntas de la evaluación deberían servir para entender qué faltó por aprender, la retroalimentación del profesor sobre las evaluaciones de la simulación, consideran debería ayudar a que el estudiante se de cuenta de los errores conceptuales y procedimentales que tiene, además están de acuerdo en que se debería evaluar los conceptos, análisis e interpretación que se trabajó en cada simulación, las guías de las simulaciones deberían presentar los contenidos conceptuales o la bibliografía necesaria para desarrollarlas. Finalmente el 44% del curso I y el 64% del curso II sugieren que las guías deberían incluir indicadores que permitieran al estudiante saber que las está desarrollando de forma correcta.

El número de horas promedio por semana que reporta el curso I que invirtió en el desarrollo de las simulaciones fue de 2.4 y en el Curso II fue de 2.8; en tanto que el número de horas promedio por semana que invirtieron los estudiantes preparando los contenidos de la asignatura de Sistemas de Comunicación fue para el Curso I de 3.9 y para el Curso II de 4.3.

IV. RESULTADOS Y CONCLUSIONES

Las 18 prácticas de simulación que se listan en la Tabla I

se aplicaron durante el primer semestre de 2008 en dos grupos de Sistemas de Comunicación I y II (comunicaciones análogas y digitales). Para el proceso de cada simulación se requiere un mínimo de tres clases, con lo cual es posible realizar cómodamente hasta 12 simulaciones (sobre lapandolas una clase entre sí) durante un semestre típico de 16 semanas. Se prevé que para el nuevo currículo (que solo tiene un curso obligatorio de Fundamentos de Comunicaciones análogas y digitales en el núcleo de formación disciplinar) se puedan realizar 12 prácticas. Igualmente se pueden usar estas simulaciones en los cursos correspondientes del énfasis en comunicaciones.

La disponibilidad de unas prácticas simuladas completas y teóricamente correctas no garantizan su éxito frente a los estudiantes: las actitudes y la motivación de los estudiantes de ingeniería electrónica frente a las temáticas de comunicaciones y sus metodologías varían muy ampliamente de acuerdo con las múltiples áreas y exigencias de esta carrera. Puesto que este curso es de carácter obligatorio, a él se presentan estudiantes con múltiples intereses, no necesariamente enfocados hacia esta área. Se debe destacar la prevención con respecto al uso directo de matemáticas (probabilidad) y las técnicas de análisis de señales, temas que pueden haber cursado con más de un semestre de diferencia con relación al curso.

Tabla I. Lista de simulaciones.

1) Procesos estocásticos: estacionariedad y ergodicidad.	9) ADPCM LMS y RLS.
2) Procesos estocásticos y sistemas lineales.	10) Modulación delta-sigma de primer orden.
3) Procesos de banda angosta.	11) Modulación delta-sigma de orden 2.
4) Modulación y demodulación de AM.	12) Receptor por correlación.
5) Modulación y demodulación de FM.	13) Señalización de respuesta parcial (duobinario).
6) Receptor FM con ruido (preénfasis y deénfasis).	14) Ecuilización adaptable.
7) Cuantificador escalar Lloyd-Max.	15) Modulación digital QAM.
8) Cuantificación robusta.	16) Modulación digital CPM.
	17) Sincronización.
	18) Codificación lineal en bloque.

La evaluación de motivaciones y actitudes nos permite determinar los ajustes necesarios para hacer que las simulaciones sean más efectivas como apoyo para el proceso de aprendizaje: para el presente semestre se prevé apoyarse en la herramienta de Blackboard para fomentar la discusión entre los estudiantes a través del foro. También se espera asegurar la ejecución oportuna de las simulaciones a través de preguntas sencillas de control del proceso que deben responderse directamente en Blackboard con anterioridad a la clase de dudas.

REFERENCIAS

- [1] Carlson, A. Bruce, Paul Crilly, Janet Rutledge, *Sistemas de Comunicación*, McGrawHill, cuarta edición, 2007.
- [2] Haykin, Simon, *Communication Systems*, John Wiley, cuarta edición, 2001.
- [3] Couch II, Leon, *Sistemas de Comunicación Digitales y Analógicos*, quinta edición, Prentice Hall, 1998.
- [4] Tomasi, Wayne, *Sistemas de Comunicaciones Electrónicas*, Pearson Education, cuarta edición, 2001.

- [5] Proakis, G., Masoud Salehi, Gerhard Bauch, *Contemporary Communication Systems Using MATLAB*, Springer; 2nd ed. edition (October 31, 2000)
- [6] Jeruchim, Michel, Philip Balaban, K. Sam Shanmugan, *Simulation of Communication Systems*, Kluber Academic/Plenum, 2000.
- [7] "Mathworks, John Hopkins University Establishes Virtual Lab for Wireless Communications with Simulink", www.mathworks.com/products/matlab/userstories.html
- [8] Kang, J., Brita Olson, Alan Felzer, Rajan Chandra, Salomon Oldak, "Simulink Based Real-Time Laboratory Course Development", Department of Electrical and Computer Engineering, California State Polytechnic University, Pomona, 2007 IEEE International Conference on Microelectronic Systems Education (MSE'07)
- [9] Krishnaswamy, Mahesh, Venkatesh A. Ramu, ECE 865 Simulink Project, <http://webscripts.softpedia.com/script/Communication-Tools/Simulink-Communication-Labs-31969.html>, 2007.

Pedro R. Vizcaya (BSc EE, P. Universidad Javeriana, Bogotá, Colombia, 1980, MSc, 1986 y PhD, 1998, Rensselaer Polytechnic Institute, Troy, NY) es profesor titular de la Facultad de Ingeniería de la P. Universidad Javeriana, con la cual se encuentra vinculado desde 1980. Es profesor del área de Análisis de Señales y Comunicaciones. Sus intereses de investigación están procesamiento de señales, reconocimiento de patrones y teoría de las comunicaciones.

Alba Lucía Meneses (BSc Psicología, Universidad Católica, Bogotá, Colombia, Msc, 1998, State University of New York, Albany, NY, DEA en Procesos Cognoscitivos y Dificultades de Aprendizaje, Universidad de Vigo, España, 2006). Profesora Universidades Católica de Colombia y El Bosque. Profesora de las asignaturas de Procesos Cognoscitivos y Metodología de la investigación. Sus intereses de investigación están en procesos cognoscitivos, lectura y escritura con estudiantes en educación superior.

Construcción y análisis no lineal de bases de datos de electrocardiogramas de felinos

Javier González B., Rafael M. Gutiérrez, Mauricio Ortega Tamayo y Leonardo Gómez.

Resumen—Categoría C4. El Electrocardiograma (ECG) es una señal de origen biológico que registra la actividad eléctrica del corazón, utilizada como herramienta en el diagnóstico y seguimiento de enfermedades del sistema cardiovascular. En el caso de poblaciones de animales domésticos (caninos y felinos), se han detectado tendencias similares en el campo de la cardiología veterinaria. En Colombia no existen trabajos de investigación que construyan y analicen bases de datos de ECG con el objetivo de ofrecer al campo de la medicina veterinaria valores de referencia de felinos de los diferentes parámetros que permitan determinar normalidad y patologías en estos animales domésticos. A través del procesamiento digital de señales de origen biológico, se pueden desarrollar herramientas informáticas que permitan el almacenamiento en forma digital del ECG libre de interferencias. Estas series de tiempo se convierten en la materia prima para el ensayo de algoritmos, basados en técnicas de análisis no lineal, con la finalidad de extraer de forma automática parámetros que faciliten la caracterización de cada registro adquirido. En este trabajo se presentará el desarrollo de una herramienta informática para la generación y análisis no lineal de bases de datos, con la capacidad de extraer de forma automática los diferentes parámetros del ECG de felinos y proponer nuevas medidas que puedan aumentar la capacidad de diagnóstico reduciendo costos y riesgos.

Palabras claves—Electrocardiograma, felinos, Herramienta Informática, Análisis no lineal.

I. INTRODUCCIÓN

Las enfermedades cardiovasculares son la primera causa de mortalidad natural humana en la mayoría de los países [1,2].

En el caso de pequeños animales existen indicios de tendencias similares [3,4,5,6]. En el diagnóstico de estas alteraciones cardiacas, el médico veterinario cuenta con el examen clínico, el electrocardiograma (ECG) y otros exámenes como la radiografía y ecocardiografía. El ECG es un registro no invasivo de la actividad eléctrica del corazón, adquirido en la superficie del cuerpo, que mide los cambios de potencial del

campo eléctrico generado por la actividad eléctrica del corazón durante las diferentes fases del ciclo cardiaco. Con las técnicas actuales, el ECG es de gran utilidad en la evaluación de la actividad eléctrica del corazón, por lo que es 99% efectiva para detectar arritmias cardiacas, pero sólo tiene un 60 a 70% de efectividad para detectar cardiomegalias [5]. El ECG puede ser adquirido a través de instrumentos electrónicos para dar los niveles necesarios a la señal para ser digitalizada [7]. Durante la adquisición del ECG es necesario tener en cuenta que pueden existir diferentes fuentes de interferencia como la alteración de la línea de base y la interferencia de 60 Hz, que dificultan el análisis de la información, por lo cual, la aplicación de la teoría de filtros digitales puede hacer confiable el análisis del registro adquirido [8, 9, 10, 11, 12, 13]. En el caso de los felinos las causas de interferencias provienen de movimientos del animal como el ronroneo y respiración [19,20].

Se han realizado diferentes trabajos en los cuales se aplican técnicas convencionales para el análisis del ECG. Muchas de estas técnicas utilizan métodos lineales de análisis como la transformada rápida de Fourier (FFT), con los cuales se puede estudiar, en el dominio de la frecuencia, arritmias cardiacas y anomalías del sistema cardiovascular [14]. El reconocimiento y caracterización de cada una de las ondas del ECG, también ha sido el tema de diversas investigaciones [15].

En el proceso de la identificación de componentes del ECG, se destaca la detección de la onda R, orientada al cálculo automatizado de los valores de frecuencia cardiaca instantánea [16]. Con los datos tomados de los tiempos entre cada una de las ondas R del complejo cardiaco, se genera una serie denominada señal de ritmo cardiaco y a través del cálculo de índices estadísticos (análisis en el dominio del tiempo) y por medio de la estimación de la densidad espectral de potencia (análisis en el dominio de la frecuencia) se puede realizar análisis de la variabilidad del ritmo cardiaco [17,18].

TABLA I
VALORES NORMALES DEL ECG EN FELINOS Y CANINOS

Especie	Duración Onda P (seg.)	Duración Seg. PR (seg.)	Duración Seg. QT (seg.)	Duración Seg. QRS (seg.)	Frec. Cardiaca (BPM)
Felinos	0,05	0,005 - 0,009	0,12 - 0,16	0,04	160 - 240
Caninos	0,04	0,06 - 0,13	0,15 - 0,25	0,06	70 - 160
		Amplitud Onda P (mV) en DII		Amplitud Onda R (mV) en DII	
Especie					
Felinos		0,2		0,9	
Caninos		0,4		0,3	

Valores normales de duración y amplitud para cada una de las ondas que componen el complejo cardiaco en felinos y caninos tomados en cuenta para el desarrollo de los algoritmos planteados en la sección II.

Artículo presentado al evento Simposio Nacional de Tratamiento de Señales, Imágenes y Visión Artificial. Celebrado en la Universidad Santo Tomás, Bucaramanga Colombia. 2008.

Javier González Barajas. Ingeniero Electrónico. Magister en Ingenierías. Docente facultad de Ing. Electrónica y Biomédica. Universidad Antonio Nariño (e-mail: javiere_gonzalez@yahoo.com.mx).

Rafael Gutiérrez Salamanca. Físico PhD. Director grupo Sistemas Complejos. DNI. Universidad Antonio Nariño. (e-mail: rafael.gutierrez@uan.edu.co).

Leonardo Gómez. Médico Veterinario. Especialista en Cardiología Veterinaria. Docente Facultad de Veterinaria. Universidad Antonio Nariño (e-mail: leonardogomez@uan.edu.co)

Mauricio Ortega Tamayo. Médico Veterinario. Especialista en Cardiología Veterinaria. Docente Facultad de Veterinaria. Universidad Antonio Nariño (e-mail: mauricioortega@uan.edu.co).

Para el caso de pequeños animales (caninos y felinos) se han realizado trabajos con el fin de establecer los valores normales de cada uno de las ondas del ECG como se puede observar en la tabla 1 [19,20].

II. MATERIALES Y MÉTODOS

El desarrollo de la herramienta informática, para la construcción y análisis de los registros electrocardiográficos de felinos, se realizó con ayuda del asistente matemático Matlab y haciendo uso principalmente de las aplicaciones especializadas en procesamiento digital de señales (Digital Signal Processing Tool Box) y algoritmos de diseño original del grupo de investigación Sistemas Complejos de la Universidad Antonio Nariño (UAN). La herramienta informática está compuesta por diversos módulos ilustrados en la figura 1.

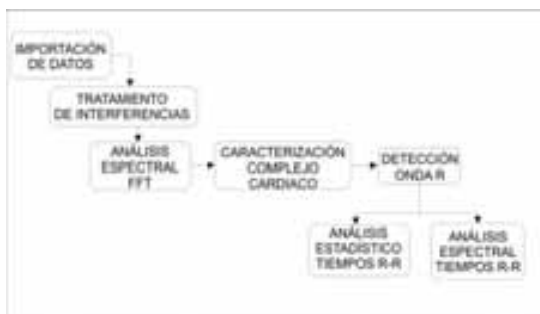


Fig. 1. Diagrama de bloques de la herramienta informática de análisis de ECG con los 6 módulos correspondientes a las etapas de procesamiento, detección y análisis.

Los registros de ECG felinos fueron adquiridos por medio del sistema de adquisición de señales biológicas del fabricante BIOPAC de referencia ECG100A. El dispositivo de adquisición utilizó una frecuencia de muestreo de 1KHz y un ancho de banda entre 0,07Hz y 40 Hz. Los registros fueron visualizados por medio del programa AcqKnowledge versión 3.2.3. Los registros adquiridos forman parte de un proyecto de investigación liderado por la Facultad de Veterinaria de la UAN con el objetivo de

TABLA II
MEDIDAS TOMADAS DEL ECG FELINO

Notación	Medida	Unidades
AP	Amplitud Onda R	mV
DP	Duración Onda P	seg
PR	Duración Seg. PR	seg
DQ	Duración Onda Q	seg
AQ	Amplitud Onda Q	mV
DR	Duración Onda R	seg
AR PRIMA	Amplitud Onda R	mV
DR PRIMA	Duración Onda S	seg
AR	Amplitud Onda R	mV
DS	Duración Onda S	seg
AS	Amplitud Onda S	mV
QRS	Duración Comp. QRS	seg
AT	Amplitud Onda T	mV
DT	Duración Onda T	seg

Medidas tradicionales que caracterizan la morfología del ECG. Estos parámetros son tomados de los complejos cardiacos de los registros adquiridos para la construcción de la base de datos de ECG felino.

construir una base de datos a partir de registros ECG de una muestra de 100 felinos. Dentro del proyecto se tienen en cuenta los parámetros de ECG ilustrados en la Tabla 2.

Tomando la estructura modular ilustrada en la figura 1, la herramienta posee el módulo Importación de Datos, el cual se encarga de tomar las series de tiempo almacenadas en archivos en formato texto de la herramienta BIOPAC. En caso de contar con registros electrocardiográficos adquiridos en papel, se ha desarrollado una estrategia basada en procesamiento digital de imágenes para digitalizar el ECG y convertirlo en una serie de tiempo, en la cual se digitaliza el registro con ayuda de un escáner y luego por medio de técnicas de erosión y binarización se extraen los píxeles que contienen la señal [21].

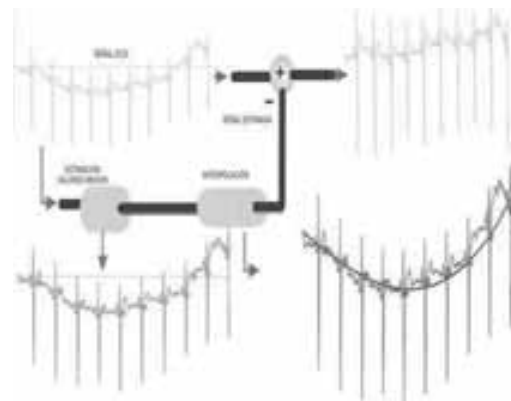


Fig. 2. Sistema de filtrado adaptativo por cancelación para eliminación Del desplazamiento de la línea de base del ECG.

Posteriormente con el fin de poder remover pequeñas interferencias, como la producida por las líneas de potencia y el desplazamiento de la línea de base producida por la respiración y ronroneo del felino [19,20], se ha desarrollado el módulo Tratamiento de Interferencias con la finalidad de minimizar estas causas de distorsiones del ECG, utilizando técnicas de filtrado adaptativo, implementando un sistema de cancelación de interferencias desarrollado en Matlab, ilustrado en la figura 2 [22].

El módulo Análisis Espectral se ha diseñado con el propósito de establecer, a través de la FFT, el rango en valores de frecuencia que cuenta con el mayor porcentaje de energía del registro almacenado. El análisis en el dominio de la frecuencia del ECG felino permitirá caracterizar cada componente de frecuencia relacionada con cada una de las ondas que componen el complejo electrocardiográfico, permitiendo adaptar las teorías de caracterización del espectro del ECG humano y generar como resultado comparaciones con el electrocardiograma de felinos [15].

El módulo Análisis Espectral estima la energía total E_T (Ver ecuación 1) del registro bajo estudio $x[n]$ que posee la cantidad de N muestras. Con base al valor E_T se estima el porcentaje acumulado de energía E_a según la ecuación 2. El cálculo de energía acumulada E_c en (2) requiere la implementación del algoritmo detallado en la figura 3, el cual estima la energía a través de la integral numérica de los valores de energía desde la componente de 0 Hz hasta un valor de frecuencia f_c que

representa la componente más alta de frecuencia para que el valor E_a alcance el 100% de la energía total.

$$E_T = \sum_{n=0}^{N-1} |X(w)|^2 \quad (1)$$

$$E_a = \frac{E_c}{E_T} \times 100 \quad (2)$$

El módulo Caracterización del Complejo Cardíaco toma la señal ECG y la divide en diferentes bandas de frecuencia, con la finalidad de poder discriminar los valores de los componentes de frecuencia que corresponden a cada onda del complejo cardíaco del ECG felino.

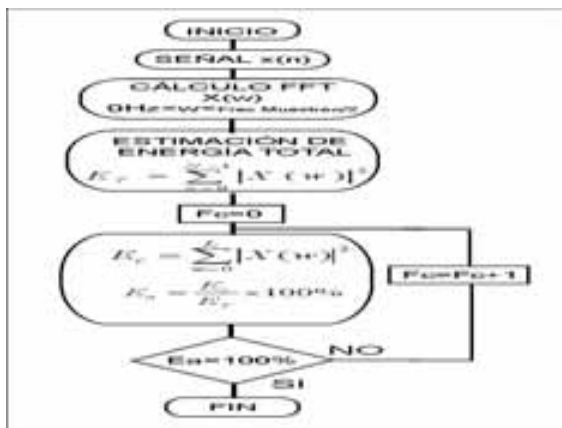


Fig. 3. Algoritmo que estima la energía acumulada de una señal $x(n)$ correspondiente a un ECG felino y el rango de frecuencias ($0 < w < ff$) que posee el mayor porcentaje

El módulo Detección Onda R es implementado con el propósito de detectar la onda R del ECG felino y cuantificar los valores de frecuencia cardíaca instantánea a través del algoritmo ilustrado en la figura 4, compuesto por un sistema de filtros en cascada (pasa bajo, pasa altas y filtro derivador) seguido por la elevación al cuadrado y un filtro integrador [15].

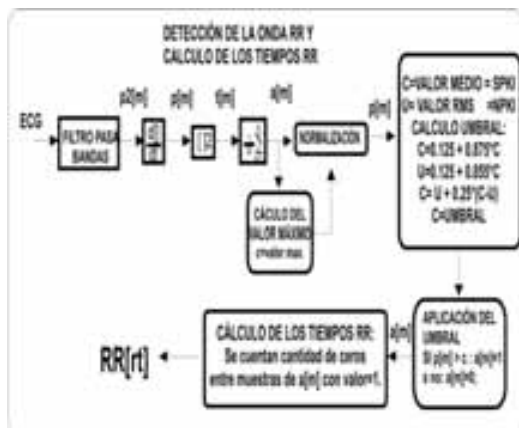


Fig. 4. Sistema para la detección de la onda R y cálculo de los tiempos RR.

módulo Detección Onda R también posee dos sub-módulos: Análisis estadístico y Análisis Espectral de los tiempos RR, que se encargan de realizar los análisis en el dominio del tiempo y de la frecuencia de los tiempos R-R respectivamente. Para el análisis en el dominio del tiempo de los intervalos RR, se calcularon los valores estadísticos descritos en la tabla 3. Con respecto al análisis en el dominio de la frecuencia, se tuvieron los valores ilustrados en la tabla 4.

TABLA III
MEDIDAS EN EL DOMINIO DEL TIEMPO

VARIABLE	UNIDAD	DESCRIPCIÓN	FORMULA
SDNN	ms	Desviación estándar de los intervalos RR	$\sigma_{NN} = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (RR_i - \bar{RR})^2}$
RMSD	ms	Raíz cuadrada del promedio de la suma de las diferencias entre intervalos RR adyacentes.	$RMSD = \sqrt{\frac{1}{N-1} \sum_{i=1}^N RR_i - RR_{i+1} ^2}$
NN50	#	Número de pares de intervalos RR adyacentes cuya diferencia es de más de 50 ms.	$(RR_i - RR_{i+1}) > 50 \text{ ms}$
pNN50	%	Porcentaje del conteo de NN50 de todos los intervalos RR (conteo de NN50/conteo total de los intervalos RR) x100.	$\% (RR_i - RR_{i+1}) > 50 \text{ ms}$

Medidas tomadas Estadísticas de los tiempos RR del ECG felino.

TABLA IV
MEDIDAS EN EL DOMINIO DE LA FRECUENCIA

Potencia VLF	m=2	< 0.04 Hz	Potencia de las componentes de muy baja frecuencia
Potencia LF	m=2	0.04 - 0.15 Hz	Potencia de las componentes de baja frecuencia
LFnu	%		Potencia LF en unidades normalizadas. Potencia LF/Potencia total - potencia VLF) x100.
Potencia HF	m=2	0.15 - 0.4 Hz	Potencia de las componentes de alta frecuencia
Hfnu	%		Potencia HF en unidades normalizadas. Potencia HF/Potencia total - potencia VLF) x100.
LF/HF			Relación de LF a HF. Potencia LF/Potencia HF

Medidas tomadas del análisis espectral de los tiempos RR del ECG felino.

III. RESULTADOS

Como resultado de este proyecto se obtuvo una herramienta informática compuesta por los módulos descritos anteriormente.

Tomando como señal de entrada un registro de ECG de felino ilustrado en la figura 5, el cual posee un desplazamiento de la línea de base. Por medio del módulo Tratamiento de Interferencias, se estiman los valores medios del registro afectado por la interferencia (ver figura 6) que son utilizados para estimar una función matemática que simule la interferencia que genera el desplazamiento de la línea de base.

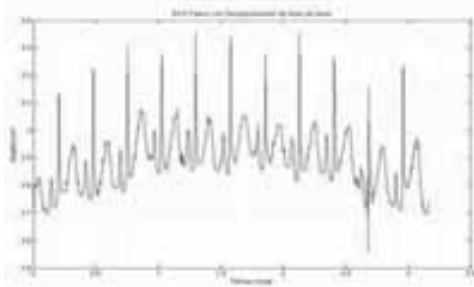


Fig 5. Electrocardiograma digitalizado de un felino y visualizado en Matlab. El registro posee desplazamiento de la línea de base.

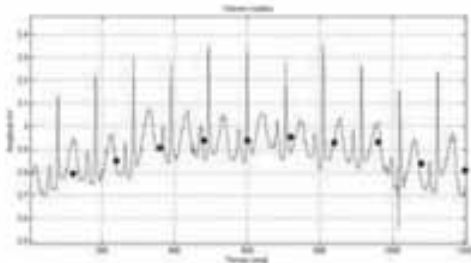


Fig 6. El módulo Tratamiento de Interferencias, estima los valores medios del registro con desplazamiento de la línea de base.

La función matemática se genera por medio de la técnica de regresión polinomial utilizando los valores medios extraídos por el módulo. En la figura 7 se puede observar la señal ECG de felino libre de desplazamiento de la línea de base, como resultado generado por el módulo de Tratamiento de Interferencias que utiliza un sistema de filtrado adaptativo por cancelación.

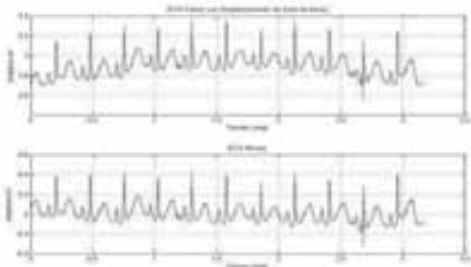


Fig 7. Resultado obtenido del proceso de remoción del desplazamiento de la línea de base de un ECG de felino realizado por el módulo Tratamiento de Interferencias.

El registro ECG libre de interferencias pasa a ser analizado por el módulo Análisis Espectral. Como resultado el módulo estima la densidad espectral de potencia a través de la transformada rápida de Fourier. La densidad espectral de potencia del registro ECG es utilizado para estimar el rango de frecuencias que contiene el mayor porcentaje de energía de la señal.

En la figura 8 se puede detallar la gráfica de energía acumulada vs frecuencia obtenida por el módulo Análisis Espectral. La figura 8 posee un eje horizontal que indica los valores de frecuencia en unidades de Hertz y el eje vertical

contiene valores de porcentaje de energía.

Por medio de esta información se establece el ancho de banda de un registro ECG de felinos. Se puede apreciar que el espectro del ECG felino acumula el 100% de energía entre las componentes de 0 y 160 Hz.

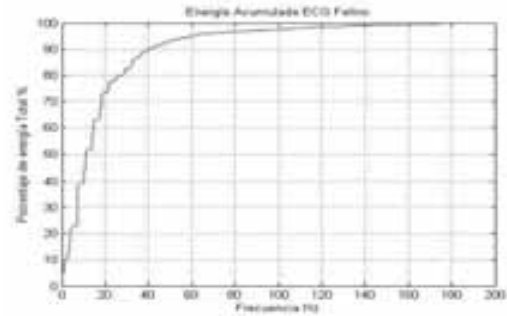


Fig 8. Cálculo del porcentaje de energía acumulada para un registro ECG de felino realizado por el módulo Análisis Espectral.

El módulo Análisis Espectral también realiza una descomposición de la señal por medio de un filtrado pasa banda con un ancho de 20 Hz. La descomposición en diferentes bandas de frecuencia del ECG Felino permite determinar el ancho de banda que está fuertemente relacionado con la onda R del complejo cardiaco. La figura 9 ilustra para ocho bandas de frecuencia diferentes la descomposición del ECG de felino con 20 Hz de ancho.

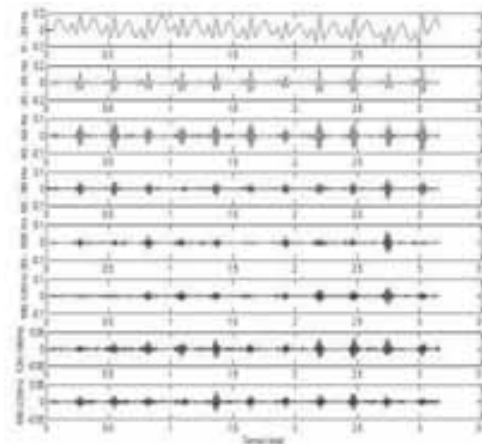


Fig 9. Descomposición del ECG felino por medio de filtros pasa banda con un ancho de 20 Hz. Para la banda de 20 - 40 Hz se resalta la onda R del complejo cardiaco.

El módulo Detección Onda R toma la señal ECG y genera el proceso para la detección de la onda R del complejo cardiaco.

Con base a la información aportada por el módulo Análisis espectral, se determina las frecuencias de corte para el filtro pasa banda (20 a 40 Hz) que se encarga de eliminar las componentes que no pertenecen al complejo QRS.

La figura 10 ilustra el resultado obtenido del proceso realizado

por el módulo Detección Onda R.

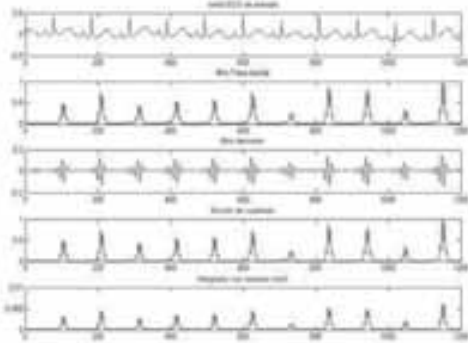


Fig 10. Proceso de detección de la onda R del complejo cardiaco de un registro ECG felino.

IV. CONCLUSIONES

La herramienta ofrece el módulo de importación de datos que permite tomar registros ECG en formato texto para su posterior análisis, lo cual ofrece la ventaja de poder manejar archivos de diferentes sistemas de adquisición de señales de origen biológico.

Los algoritmos implementados para el tratamiento del desplazamiento de la línea de base del ECG en el módulo Tratamiento de Interferencias, son utilizados tradicionalmente en el procesamiento de registros electrocardiográficos de humanos. En este artículo se muestran resultados positivos de la aplicación de estos algoritmos para el caso del ECG de felinos. Aquí cabe notar la dificultad adicional que significa analizar ECG de felinos por su mayor frecuencia cardiaca y menor amplitud relativa, dos características que hacen más difícil la separación de la señal de interés de las diferentes fuentes de energía.

El módulo Análisis Espectral permitió obtener una medida que indica las componentes en frecuencia que contiene el mayor porcentaje de información del ECG felino, las cuales están entre $0 < f_c < 160$ Hz.

Finalmente como perspectiva, esta herramienta permitirá analizar una base de datos de ECG felino para construir una tabla de valores de referencia como se ilustró en la tabla II, que caracterice la morfología de ECG felino y ofreciendo a la medicina veterinaria una referencia útil en el diagnóstico de patologías y normalidad cardiaca en felinos.

REFERENCIAS

- [1] J.S. Alpert, Coronary Heart disease: where have we been and where are we going?, *The Lancet* V 353, p.1540 (1999).
- [2] A.J.S Coats, Is preventive medicine responsible for the increasing prevalence of heart failure, *Lancet* V 352 (suppl)p.39 (1998).
- [3] A. dic, R Langenford. Interpretación de arritmias complejas. Intermédica (1984).
- [4] R. Kira, Terapéutica veterinaria de pequeños animales, Interamericana 11ªed, Madrid, p.1409 (1994).
- [5] G. Belerenian, Afecciones cardiovasculares en pequeños animales, Intermédica, Buenos Aires, p.38 (2001).
- [6] V Luís Fuentes, S.T.Swift, Manual de medicina y cirugía cardiorrespiratoria, Hartcourt, p. 345 (2000).

- [7] Tompkins, Willis. Biomedical Digital Signal Processing. Prentice Hall. New Jersey. 1993. 367:p24-54.
- [8] Shen P, Lindquist C. Adaptive Filtering of EKG Signal with Little A priori Information. Department Of Electrical Engineering. Santa Clara University. IEEE, 2000 0-783-6514-3/00 : 338-342.
- [9] C.S. Lindquist. Adaptive and Digital Signal Processing. Steward ans Sons. CA. 1989.
- [10] Shen P, Lindquist C. Adaptive Filtering of EKG Signals With Little Apriori Information. IEEE 2000. 0-7803-6514-3/00/\$10.00.
- [11] Bellanger M. Digital Signal Processing of Signals. Theory and Practice. John Wiley and Sons. 1994. P20.
- [12] Jané R, Laguna P, Caminal P, Rix Hervé. Adaptive Filtering of High Resolution ECG Signal. IEEE. USA 1991. 0276-6574/91/00000/0347\$01.00. P347 – P50.
- [13] Laguna P, Jané R, Caminal P. Adaptive Filtering of ECG Baseline Wnader. IEEE 1992. 0-7803-0785-2/92\$03.00.
- [14] Pfister, C.J., Hamilton J.W. Use of Spectral Analysis in Detection of Frequency Differences in Electrogastragram 1998, IEEE Trans. Biomed Eng. 935-41.
- [15] Thakor, N.V. Webster JG and Tompkins, WJ. Estimation of QRS complex Power Spectra for Desing of a QRS Filter. 1984. IEEE Trans. Biomed. Eng., BME-31:702-05.
- [16] Thakor, N.V. Webster JG and Tompkins, WJ. Optimal QRS Detector. Medical and Biological Engineering. IEEE Trans. Biomed. Eng., BME-31: 1984. 343-50.
- [17] Robert, E. Kleiger MD, Facc, Phyllis K. Stein, PhD. Time Domain Measurements of Heart Rate Variability. Ambulatory Electrocardiography. Cardiology Clinics. Washington University Medical Center. St. Lius. Missouri. USA. 10:p487-p489.
- [18] ZSOLT, Ori, MD. Heart Rate Variability. Frequency Domain Analysis. Ambulatory Electrocardiography. Northwestern University Medical School, Chicago. USA August 1992. 10: p499 – p501.
- [19] Tilley L, Burtmick N. Electrocardiografía Práctica en Pequeños Animales. Manual de Consulta Rápida. Multimédica ed. España. 2000. 97: p16-p17.
- [20] Tilley L, Owens J. Manual de Cardiología de los Pequeños Animales. Salvat Editores. España 1987. 427: p57.
- [21] Rosas A, Plata A, Chávez AM. Obtención de patrones electrocardiográficos de una ballena jorobada mediante tratamiento de imagen y señal. Rev. Col. Cardiol. v.12 n.4 Bogota dic. 2005.
- [22] González BJ y Barrero JG. Implementación de Filtros Adaptativos en DSP aplicados al Tratamiento de Interferencia de 60 Hz y Desplazamiento de la línea de Base. Energía y Computación. Volumen XIII Número 2 Edición 24. Pag 16-19.

Sistema de reconocimiento y lectura de placas de vehículos en movimiento

Vivianne A. Mahecha, Ing. Julián Quiroga Sepúlveda M. Sc

Resumen—Este trabajo presenta un sistema de reconocimiento y lectura de placas de vehículos en movimiento desarrollado para placas colombianas. El sistema localiza objetos que concuerden con las características de una placa en un cuadro de video, verifica y segmenta dichos objetos, realiza un realce a la imagen segmentada (la placa) y por último efectúa el reconocimiento de caracteres. El sistema implementado es probado bajo diferentes condiciones de adquisición.

Palabras Claves—OCR (Optical Character Recognition), Plano de color YCrCb, Niblack, operador Sobel, Histograma, Operaciones morfológicas: Erosión, Dilatación; OpenCV.

I. INTRODUCCIÓN

El reconocimiento automático de placas o mejor conocido como ANPR por sus siglas en inglés *Automatic Number Plate Recognition*, se le atribuye a la compañía inglesa *Police Development Branch* en 1976. Este sistema consiste en la adquisición de imágenes del tráfico bien sea por un circuito cerrado de televisión -CCTV- o por cámaras especializadas, a las cuales se les hace un pre-procesamiento generalmente compuesto por tres pasos principales: localización de la placa dentro de la imagen, normalización de contraste y brillo y por último segmentación de caracteres; una vez terminado, la imagen queda lista para pasar por un sistema de reconocimiento de caracteres OCR – *Optical Character Recognition*.

Son muchas las aplicaciones que un sistema de ANPR puede brindar, entre las más empleadas se encuentran: el registro automático de vehículos a la entrada y salida de un parqueadero las 24 horas del día; el mejoramiento de la movilidad en áreas de congestión como en Londres en donde existe una multa para ciertos vehículos al entrar en áreas catalogadas como áreas de congestión; permitir el acceso de vehículos autorizados; proveer información acerca del flujo vehicular y el paso automático de vehículos autorizados por peajes.

Este trabajo consiste en desarrollar e implementar un algoritmo que reconozca y realice la lectura de placas de vehículos en movimiento a partir de un cuadro de video. Para tal fin se deberá sobrepasar dificultades producidas por el ángulo de captura, el ángulo de inclinación de la cámara, la

velocidad de los vehículos, la calidad y resolución de la imagen, diferentes factores de ruido en el objeto de interés como árboles, peatones, bicicletas, otros carros etc.

A continuación se muestra de forma clara y detallada las especificaciones del sistema, así como los requerimientos que deben ser satisfechos por el mismo, indicando las operaciones que debe realizar, sus entradas y salidas; también se presenta una descripción completa de la organización global y de la estructura del sistema; y un desarrollo detallado de su funcionamiento.

II. ESPECIFICACIONES

El sistema de reconocimiento y lectura de placas de vehículos en movimiento puede ser descrito como un solo bloque el cual tiene como única entrada un cuadro de video y puede presentar de cero a dos salidas, dependiendo del número de placas que se hayan encontrado en la imagen. Figura 1.

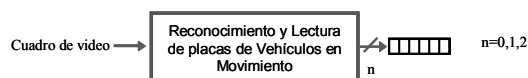


Figura 1. Entradas y salidas del sistema

La imagen de entrada, como se dijo anteriormente, es un cuadro de video que pertenece a una de las cuatro tomas realizadas. Estas tomas fueron hechas a veinte, cuarenta, sesenta y ochenta grados tomando como cero la recta perpendicular a la vía, como se muestra en la Figura 2. La vía elegida fue la carrera 11 con calle 84 de la ciudad de Bogotá, la cual presenta un flujo vehicular alto con velocidad promedio aceptable (aprox. 27 km/h) y un número bajo de congestiones al no permitir el tráfico de vehículos de servicio público masivo, delimitando el problema a la ubicación de placas de carros particulares y de servicio público (taxis).

El cuadro de video pasa por una primera etapa de localización de candidatos, los candidatos siguen a una etapa de verificación en donde se determina cual de estos corresponde a una placa. Después de haber hallado las placas en la imagen, se les aplica un pre-procesamiento para realizar

la imagen y así dejarla lista para el reconocimiento de caracteres.

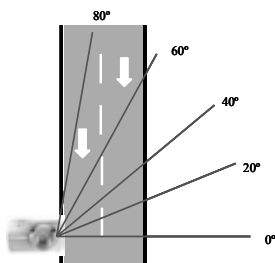


Figura 2. Ángulos de toma

La o las salidas son vectores de seis caracteres alfa numéricos con las letras y números de la o las placas encontradas.

III. DESCRIPCIÓN Y DESARROLLO

El sistema esta dividido en cuatro bloques generales, *Búsqueda de candidatos*, *Verificación y segmentación*, *Pre-procesamiento*, y *Reconocimiento de caracteres*, como se muestra en la Figura 3.

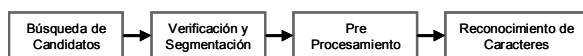


Figura 3. Diagrama en bloques general

A. Búsqueda de candidatos.

Como se muestra en la Figura 4, este bloque tiene como entrada un cuadro de video y el número de salidas varia dependiendo del número de candidatos encontrados en la imagen que cumplan con características mínimas de color y área de una placa, las cuales son expuestas a continuación.

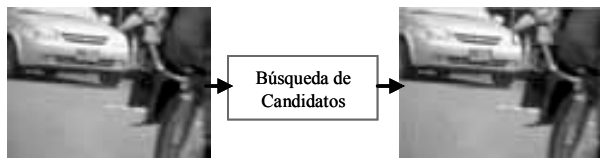


Figura 4. Búsqueda de Candidatos

El bloque de *Búsqueda de candidatos* se divide en dos sub-bloques conectados en paralelo, como se ilustra en la Figura 5. Cada uno de estos corresponde a la búsqueda de candidatos de los dos tipos de placas presentes.

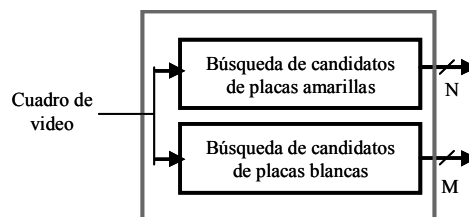


Figura 5. Búsqueda de candidatos

1) Búsqueda de candidatos de placas amarillas

Una imagen en el plano Cb muestra los objetos que carecen en gran medida del color azul, como los objetos amarillos o muy cercanos a esta tonalidad. Dado lo anterior, la búsqueda de placas amarillas inicia en el plano de color Cb, una vez ubicados en este plano se hace un realce en el que se intensifica el contraste para lograr una mejor *umbralización* con el objetivo de mostrar como verdaderos los píxeles correspondientes a objetos amarillos.

2) Búsqueda de candidatos de placas blancas

Siguiendo con la justificación presentada en el bloque anterior respecto al plano Cb, se realiza el mismo procedimiento con la diferencia que la *umbralización* muestra como falsos los píxeles correspondientes a objetos con bajo nivel de azul, esto con el fin de rechazar los vehículos de color amarillo - taxis-. Paralelo a esto la imagen pasa por un bloque en el que se determinan los bordes verticales con ayuda del operador Sobel queriendo así, con un proceso de dilatación, unir los bordes que estén suficientemente cerca (placa). Estos dos resultados se intersectan dejando los lugares con componentes de azul elevadas (blancos) y bordes verticales cercanos.

Teniendo ya la imagen con los posibles candidatos, ésta pasa por un filtro el cual rechaza áreas menores a 1000 píxeles con el fin de eliminar el ruido; este limite se determinó como conclusión de la observación de la base de datos, que recordando, esta constituida por diferentes ángulos de captura en donde se presentan placas entre 1323 y 29512 píxeles de área dependiendo del lugar en el que se encuentre el carro dentro de la imagen y del ángulo de captura.

B. Verificación y Segmentación.

Las salidas del bloque anterior se convierten en las entradas de éste. El bloque verifica la elección de candidatos presentada según criterios más específicos en base a características propias de una placa. Una vez hecha esta verificación el bloque extrae dichos objetos mostrando a la salida imágenes aisladas de la(s) placas(s) encontrada(s). Dependiendo del ángulo de captura se pueden encontrar hasta dos placas por cuadro de video, como se ve en la Figura 6.

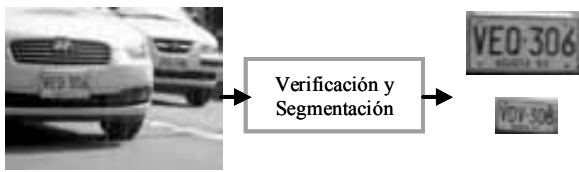


Figura 6. Imagen de entrada y placas segmentadas.

Este bloque se divide en tres etapas conectadas en serie, descritas a continuación. Figura 7:

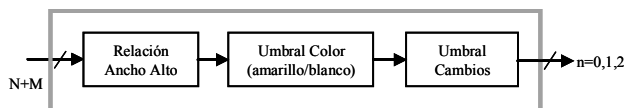


Figura 7. Verificación y segmentación

1) Relación ancho y alto

Según la morfología de una placa que corresponde a un rectángulo horizontal, los candidatos pasan por un filtro que evalúa la relación ancho y alto de las áreas seleccionadas por el bloque anterior, esta relación debe estar dentro de un intervalo específico, 0.6 - 2.3, que fue determinado por observación al implementar el sistema en 500 placas. Esta característica, al igual que la característica de área descrita anteriormente, depende del ángulo de captura y ubicación del carro dentro de la imagen, por lo tanto los límites corresponden a las tomas de 20° y 60° respectivamente en donde se encuentran los casos límite de esta característica entre relación ancho y alto.

2) Umbral Color

Esta etapa verifica y determina si los candidatos son objetos en los cuales predomina bien sea el color amarillo o el color blanco. Para esto se evaluó, en el caso de las placas amarillas, que cierto porcentaje del histograma de los planos verde (G) y azul (B) estuviese contenido en umbrales predeterminados. En el caso de las placas blancas se evaluó la desviación estándar del umbral, en los tres planos de color rojo, verde y azul, en el que el histograma alcanza el 50%.

3) Umbral Cambios

Los candidatos restantes son *binarizados* con la técnica de *binarización* adaptativa de Niblack, esto con el fin de determinar el número de cambios entre negro (fondo) y blanco (números o letras) que se presenta en la mitad de la imagen. Teniendo esta información los candidatos pasan por un filtro que rechaza las imágenes con cambios inferiores a 8 y superiores a 28. Este rango se determinó pensando en el peor de los casos, el rango inferior corresponde a una placa compuesta de solo 0 o 1 en donde los 6 caracteres, sin contar los bordes de la placa y asumiendo que dos de estos se ven unidos por la baja calidad de la imagen, presentarían solo

ocho cambios. El rango superior corresponde a una placa en donde cada caracter tiene asociado 4 cambios –por ejemplo una U o un 4– se tiene en cuenta el cambio en el borde de las placas y los 2 cambios producidos por el logotipo del ministerio de transporte en la mitad de la placa.

C. Pre-Procesamiento

De acuerdo al trabajo desarrollado por Shaefer [7], se implementó la técnica de deformación de imágenes ya que, como consecuencia de los diferentes ángulos de captura las placas requieren de cierta deformación o rotación para ser leídas. La Figura 8 muestra dos ejemplos de placas por mejorar.



Figura 8. Placas por mejorar

Esta técnica de deformación consiste en definir unos puntos de control que serán movidos a nuevos puntos en la imagen deformada, en la Figura 9 se puede apreciar un ejemplo en donde los puntos verdes representan los puntos de control (p_i) y los puntos azules representan los puntos deseados (q_i).



Figura 9. Puntos de control y puntos deseados

Teniendo en cuenta esto se define una función f que mueva cada píxel de la imagen original a la imagen deformada, dicha función debe cumplir las siguientes propiedades.

- Si el píxel por mover pertenece a uno de los puntos de control (p_i), entonces la función debe interpolarlos a los puntos deseados (q_i): $f(p_i) = q_i$
- Si los puntos deseados resultan ser los mismos que los puntos de control, la función f debe comportarse como la función identidad. $f(v) = v$, siendo v cualquier píxel dentro de la imagen.

- La función f debe producir deformaciones suaves.

La función con la cual se trabajó fue la definida en Shaefer [7], la cual corresponde a una deformación “afín”. En la Figura 10 se puede ver el resultado de la implementación de esta técnica.

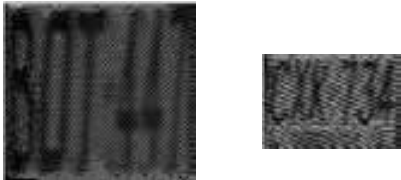


Figura 10. Placas mejoradas

D. Reconocimiento de caracteres

Una vez listas las placas para ser leídas gracias al pre-procesamiento realizado, el bloque de reconocimiento de caracteres tiene como primera función el de segmentar y escalizar dichos caracteres.

Cada uno de los caracteres será evaluado por separado por un algoritmo de OCR. Actualmente se encuentra en desarrollo el algoritmo de OCR el cual se basa en redes neuronales tomando como características de cada letra las evaluadas e implementadas por Frey [10].

IV. CONCLUSIONES

Desarrollar un sistema de reconocimiento y lectura de placas de vehículos en movimiento como este, en donde se pone a prueba el sistema bajo diferentes ángulos de captura, permite identificar las características relevantes que afecten en el algoritmo de reconocimiento de caracteres. Dando así como resultado un trabajo en donde no solo se evalúa la efectividad del sistema sino que también se brindan las condiciones óptimas para su implementación.

REFERENCIAS

- [1] R. Gonzalez, R. Woods. Digital Image Processing Using Matlab. Ed Person Education New Jersey 2004.
- [2] R. Gonzalez, R. Woods. Digital Image Processing. Ed. Prentice Hall Massachusetts 1992.
- [3] Xiang Pan, Xiuzi Ye, Sanyuan. A Hybrid Method for Car Plate Character Recognition. IEEE International Conference on System, Man and Cybernetics. 2004
- [4] Chen Qing. Evaluation of OCR Algorithms for Images with Different Spatial Resolution and Noise. Master of Applied Science Thesis. 2003

[5] Taxt Torfinn, Trier. Evaluation of Binarization Methods for Document Images. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 17 No 3. March 1995.

[6] Blayvas Ilya, Bruckstein Alfred, Kimmel Ron. Efficient computation of adaptative threshold surfaces for image Binarization. Pattern Recognition, The journal of the pattern recognition society. 2005.

[7] Shaefer Scott, McPhail Travis, Warren Joe. Image Deformation Using Moving Least Squares.

[8] BBC News. Met given real time c-charge data. 2007 July. Disponible en: http://news.bbc.co.uk/2/hi/uk_news/politics/6902543.stm

[9] Plate Recognition. <http://www.photocop.com/recognition.htm>

[10] Frey Meter, State David. Letter Recognition Using Holland-Style Adaptive Classifiers. Machine Learning Vol 6. 1991.

Evaluación de diferentes matrices de medición utilizando orthogonal matching pursuit (OMP)

Adriana Lobato, Rafael Ruiz, Adolfo Recio y Julián Quiroga

Resumen— Muestreo Compresivo es una rama emergente del procesamiento de señales, basada en el hecho de que un número pequeño de proyecciones lineales no adaptativas sobre una señal compresible, contienen suficiente información para reconstruirla y procesarla. En este artículo se presentan los resultados obtenidos al evaluar cinco matrices de medición para la realización de Muestreo Compresivo en un sistema que utiliza el algoritmo Orthogonal Matching Pursuit (OMP), para la recuperación de la señal original. Las matrices de medición están implicadas tanto en el proceso de muestreo-compresión de la señal, como en la reconstrucción de la misma.

Palabras claves— Muestreo Compresivo, algoritmo Orthogonal Matching Pursuit, matriz de medición.

I. INTRODUCCIÓN

MUESTREO Compresivo (MC) es un enfoque alternativo y optimizado al principio planteado por el teorema de Shanon/Nyquist, el cual permite, potencialmente, el desarrollo de aplicaciones a nivel de hardware y software menos costosas. Utilizando el principio de MC es posible realizar dos procesos de forma simultánea en la adquisición de la información: sensar y comprimir. La tecnología convencional, en contraste, realiza un proceso de muestreo seguido por uno de compresión.

Una señal dispersa o *sparse* es aquella que sólo presenta pocos valores distintos de cero en el dominio del tiempo o en algún otro dominio. El número de valores no nulos de una señal dispersa se conoce como su nivel de dispersión o *sparcity*. Muchas señales tienen representaciones dispersas en cierto espacio. Una señal perteneciente a un espacio vectorial tiene una representación dispersa, si ésta puede ser expresada como una combinación lineal de pocos vectores de una base de dicho espacio. Por ejemplo, las señales formadas por componentes armónicas son dispersas en bases de Fourier, y las suaves a trozos tienen representaciones wavelets con estructuras dispersas [1],[2],[3].

La matriz de medición o diccionario permite capturar información de una señal dispersa dada y es utilizada por el algoritmo OMP para la recuperación de dicha señal una vez ha sido comprimida. El hecho de que la matriz de medición presente un pequeño número de filas (vectores de medición) tiene ventajas como mayor compresión de la información y menor tiempo de procesamiento. Por tal motivo, en este artículo se presenta el estudio de diferentes clases de matrices de medición, con el fin de encontrar aquellas que permiten

obtener una recuperación adecuada de la señal utilizando un menor número de vectores de medición.

En la Sección 2 se presentan algunos conceptos básicos necesarios para la teoría de MC y se define parte de la notación utilizada. En la Sección 3 se define el problema de recuperación de una señal que ha sido comprimida bajo el principio de MC y se presenta OMP como solución a dicho problema.

En la Sección 4 se detalla la metodología y las herramientas empleadas para la evaluación de las matrices, basadas en mediciones estadísticas. En la Sección 5 se describe cada una de las matrices de medición propuestas. En la Sección 6 se muestran los resultados más relevantes del trabajo y se realiza un análisis de los mismos. En la Sección 7 es llevado a cabo una breve comparación con trabajos precedentes. Finalmente, en la Sección 8 se presentan algunas conclusiones del trabajo realizado.

II. MUESTREO – COMPRESIÓN DE SEÑALES

A. Muestreo Compresivo

Según el principio de Muestreo Compresivo, una señal discreta con una representación dispersa en cierta base, puede ser recuperada a partir de un número pequeño de proyecciones lineales de dicha señal sobre una base arbitraria [4].

MC sugiere una nueva forma de adquirir la información que necesita ser convertida de análoga a digital, debido a que los procesos de muestreo y compresión de la señal se realizan simultáneamente [8]. Cuando se pretende aplicar el teorema de Shannon/Nyquist a sistemas de radio-frecuencia de amplio ancho de banda se presentan limitaciones de *hardware* para la implementación de una etapa análogo-digital de alta precisión y alta velocidad. Haciendo uso del principio de MC no sería necesario que todas estas muestras pasaran por dicho conversor, sino sólo un pequeño número de observaciones de la señal, lo que reduciría los requerimientos de velocidad del conversor.

B. Compresión – muestreo de la señal

Sea $\mathbf{s} \in \mathbf{R}^d$ una señal discreta real de longitud d , representada como un vector columna de d posiciones (1),

$$\mathbf{s} = [s_1 \quad s_2 \quad \dots \quad s_d]^T \quad (1)$$

Sea $\Phi = [\varphi_1 \quad \varphi_2 \quad \dots \quad \varphi_d]$, con $\varphi_i \in \mathbf{R}^N$ para $i = 1, 2, \dots, d$. La matriz Φ es un elemento de $\mathbf{R}^{N \times d}$ y se

conoce como matriz de medición. Sus columnas, los vectores Φ_i , se denominan átomos, los cuales tienen norma unitaria.

Cada una de las N filas de Φ se conoce como vector de medición. El proceso de muestreo-compresión consiste en la obtención del vector \mathbf{v} como la combinación lineal de átomos de Φ dada por:

$$\mathbf{v} = \sum_{i=1}^d s_i \Phi_i = \Phi \mathbf{s}, \quad (2)$$

donde s_i es el i -ésimo elemento de \mathbf{s} .

El vector columna $\mathbf{v} \in \mathbf{R}^N$, se conoce como vector de observación. Éste contiene información importante acerca de la señal \mathbf{s} .

El tamaño del vector \mathbf{v} resultante, depende de N , a continuación analizamos dos casos de interés. En el primero de ellos, si $N = d$, no existe propiamente compresión de los datos pues $\mathbf{v} \in \mathbf{R}^d$, es decir, la señal original y el vector \mathbf{v} tienen la misma longitud. En el segundo caso, si $N < d$, la dimensión del vector \mathbf{v} es menor a la dimensión de \mathbf{s} y por tanto, existe compresión de la información. Se puede demostrar que bajo ciertas condiciones de \mathbf{s} y de Φ , la información contenida en \mathbf{v} puede ser suficiente para recuperar de forma aproximada la señal \mathbf{s} .

Si \mathbf{s} es una señal m -dispersa, es decir, solamente m de sus elementos son distintos de cero, la ecuación (2) equivale a una combinación lineal de sólo m átomos de Φ , es decir

$$\mathbf{v} = \sum_{s_i \in \sigma} s_i \Phi_i$$

donde $\sigma = \{s_i \in \mathbf{s} : s_i \neq 0\}$ y $|\sigma| = m$.

III. RECUPERACIÓN DE SEÑALES DISPERSAS

Una vez se ha realizado la combinación lineal de átomos de Φ , el objetivo es recuperar la señal original \mathbf{s} a partir del vector de observación \mathbf{v} . La recuperación difiere en complejidad dependiendo de N , a continuación se analizan los dos casos mencionados en la Sección 2.

Para el primer caso, si $N = d$ y los átomos de Φ constituyen un conjunto ortonormal (elementos mutuamente ortogonales y con norma unitaria), la señal \mathbf{s} puede ser recuperada como:

$$s_i = \langle \mathbf{v}, \Phi_i \rangle, \quad i = 1, 2, \dots, d$$

donde la notación $\langle \cdot, \cdot \rangle$ indica el producto interno en el espacio \mathbf{R}^d .

En este caso la representación del vector \mathbf{v} sobre los átomos de Φ permite recuperar de forma unívoca la señal \mathbf{s} .

Para el segundo caso y el de mayor interés, en el cual $N < d$, los átomos de Φ no forman un conjunto linealmente independiente y por lo tanto tampoco un conjunto ortogonal. Por tal motivo, la recuperación de la señal no puede ser realizada descomponiendo el vector \mathbf{v} como un conjunto de proyecciones sobre los átomos de Φ .

Uno de los métodos para recuperar la señal \mathbf{s} a partir del

vector \mathbf{v} [5], consiste en encontrar la señal $\hat{\mathbf{s}}$ que minimiza la energía del error entre \mathbf{v} y la combinación lineal de átomos de Φ utilizando $\hat{\mathbf{s}}$, es decir, resolver el problema de optimización:

$$\min_{\{\hat{\mathbf{s}}\}} \left\| \mathbf{v} - \sum \hat{s}_i \Phi_i \right\|^2 \quad (3)$$

donde $\|\cdot\|$ es la norma euclidiana en \mathbf{R}^N , definida como la raíz del producto interno y \hat{s}_i , para $i = 1, 2, \dots, d$, corresponde al i -ésimo valor de $\hat{\mathbf{s}}$.

Si la señal \mathbf{s} es m -dispersa se deben identificar los m átomos de Φ que participan en la combinación lineal para formar \mathbf{v} . En este caso se debe encontrar la señal $\hat{\mathbf{s}}$ m -dispersa que minimiza la ecuación (3).

El éxito de la compresión y recuperación de la señal \mathbf{s} , depende del balance existente entre los siguientes tres factores:

1. La relación N/d que determina el grado de compresión de la información.
2. La matriz Φ .
3. El nivel de dispersión de la señal \mathbf{s} .

Una solución al problema de optimización (3) es el algoritmo Orthogonal Matching Pursuit (OMP).

A. Proceso iterativo

OMP es un algoritmo que en cada iteración selecciona la opción óptima a nivel local, dentro de un conjunto de posibles soluciones, esperando encontrar al final la solución óptima global.

El propósito de OMP es realizar una aproximación del vector \mathbf{v} por medio de la selección y combinación de un conjunto de átomos de Φ que minimizan (3) en cada iteración.

La aproximación de \mathbf{v} se construye mediante un proceso iterativo, para lo cual el algoritmo selecciona en cada paso el átomo de Φ mejor correlacionado con el residuo de la señal [9], el cual está definido como la diferencia entre \mathbf{v} y su aproximación parcial, y se denota con la letra \mathbf{r} .

Para indicar la iteración correspondiente se utiliza el subíndice t .

Al comienzo del proceso iterativo ($t = 0$), el residuo \mathbf{r}_0 es igual al vector \mathbf{v} . Seguidamente, se busca la máxima proyección de \mathbf{r}_0 sobre el conjunto de átomos de la matriz de medición.

Debido a que cada uno de los átomos se encuentra normalizado, el átomo que presenta la mayor proyección es encontrado como

$$j_0 = \arg \max_{j=1,2,\dots,d} \left| \langle \mathbf{r}_0, \Phi_j \rangle \right| \quad (4)$$

$$\lambda_0 = \langle \mathbf{r}_0, \Phi_{j_0} \rangle \quad (5)$$

En (4) se determina la posición j_0 del átomo que proporciona el mayor producto interno, el cual se conoce como átomo

escogido Φ_{j_0} . El producto interno máximo dado por Φ_{j_0} y \mathbf{r}_0 se almacena en la variable λ_0 .

Posteriormente se procede a calcular el nuevo residuo \mathbf{r}_1 , como la diferencia entre el residuo anterior y la proyección de éste sobre el átomo seleccionado

$$\mathbf{r}_1 = \mathbf{r}_0 - \langle \mathbf{r}_0, \Phi_{j_0} \rangle \Phi_{j_0} \quad (6)$$

Claramente Φ_{j_0} es ortogonal a \mathbf{r}_1 , por lo tanto

$$\|\mathbf{r}_1\|^2 = \|\mathbf{r}_0\|^2 - \left| \langle \mathbf{r}_0, \Phi_{j_0} \rangle \right|^2$$

Para minimizar la norma del nuevo residuo (error) $\|\mathbf{r}_1\|^2$, el algoritmo OMP escoge el átomo Φ_{j_0} que maximiza

$$\left| \langle \mathbf{r}_0, \Phi_{j_0} \rangle \right|.$$

Este procedimiento se repite hasta seleccionar m -átomos de Φ . Con el fin de evitar la selección repetida, el átomo escogido en la iteración t , Φ_{j_t} , se ortogonaliza con los átomos escogidos en iteraciones pasadas y que han sido previamente ortogonalizados. Para dicho fin se utiliza el procedimiento de Gram-Schmidt, de acuerdo a

$$\Phi'_{j_t} = \Phi_{j_t} - \sum_{l=0}^{t-1} \frac{\langle \Phi_{j_t}, \Phi'_{j_l} \rangle}{\|\Phi'_{j_l}\|^2} \Phi'_{j_l} \quad (7)$$

donde Φ'_{j_t} es la versión ortogonalizada del átomo escogido en la iteración t .

Con el átomo Φ'_{j_t} se calcula el vector residuo de la iteración t (8).

$$\mathbf{r}_t = \mathbf{r}_{t-1} - \frac{\langle \mathbf{r}_{t-1}, \Phi'_{j_t} \rangle}{\|\Phi'_{j_t}\|^2} \Phi'_{j_t}, \quad t = 2, \dots, m-1 \quad (8)$$

En otras palabras la actualización del residuo consiste en suprimir del residuo actual la proyección que tiene éste sobre el átomo escogido ortogonalizado (9).

$$\mathbf{r}_t = \mathbf{r}_{t-1} - P_{\Phi'_{j_t}} \mathbf{r}_{t-1} \quad (9)$$

Sean \mathbf{J} y Λ los conjuntos que almacenan para cada iteración, las posiciones de los átomos escogidos y los máximos productos internos, respectivamente, es decir

$$\mathbf{J} = \{j_0, j_1, \dots, j_{m-1}\}$$

$$\Lambda = \{\lambda_0, \lambda_1, \dots, \lambda_{m-1}\}$$

En cada una de las iteraciones, se adjuntan elementos al conjunto \mathbf{J} y Λ como en (4) y (5).

La señal \mathbf{s} se reconstruye al final de las iteraciones como en (10), en donde se observa que la amplitud está dada por los

elementos del conjunto Λ cuyas posiciones en el vector $\hat{\mathbf{S}}$ están determinadas por los elementos del conjunto \mathbf{J} . Es decir, la posición j_i en $\hat{\mathbf{S}}$ contiene la amplitud λ_i .

$$\hat{\mathbf{S}}(\mathbf{J}) = \Lambda \quad (10)$$

IV. HERRAMIENTAS DE EVALUACIÓN

Se realizó una rutina utilizando MATLAB® con la que se obtuvieron los resultados presentados a continuación. El algoritmo se ejecutó con 1000 señales aleatorias para cada número de vectores de medición N . Las características de la señal \mathbf{s} son:

- Vector de 256 posiciones.
- Amplitud uniformemente distribuida en el intervalo de enteros [-10,10].
- Niveles de dispersión: 4, 12, 20, 28, 36. que corresponden a los mismos niveles de dispersión empleados en [6].
- Las posiciones no nulas de la señal se encuentra uniformemente distribuidas en el intervalo de enteros [1,256].

El *error de aproximación* (e), es una medida de la exactitud entre la señal recuperada y la señal original y se define como $\|\mathbf{s} - \hat{\mathbf{s}}\|$ [5].

Definamos el *porcentaje de recuperación* (PR) de la señal como:

$$\text{PR}(e) = \begin{cases} \left(100 - \frac{100e}{\|\mathbf{s}\|}\right) \% & , \text{ si } 0 \leq e \leq \|\mathbf{s}\| \\ 0 \% & , \text{ si } e > \|\mathbf{s}\| \end{cases}$$

El error de aproximación y el porcentaje de recuperación brindan información importante sobre el desempeño del algoritmo, sin embargo utilizamos el segundo tipo de medición, pues éste es independiente de la norma de \mathbf{s} . Un PR de 100% implica un error cero, mientras un PR de 0% implica un error mayor o igual a $\|\mathbf{s}\|$.

Con base en los valores del porcentaje de recuperación de las señales en función de N , se obtuvieron gráficas estadísticas e histogramas tridimensionales.

V. MATRICES DE MEDICIÓN PROPUESTAS

La matriz aleatoria Gaussiana, como su nombre lo indica, contiene números aleatorios con distribución Gaussiana. La matriz se generó a partir de la función *randn* de MATLAB®, con media cero y desviación estándar unitaria.

Para la matriz Bernoulli, se implementó un caso particular de esta distribución, conocida con el nombre de distribución de Rademacher, cuya función de densidad de probabilidad es

$$f(k) = \begin{cases} 1/2 & \text{para } k = 1 \\ 1/2 & \text{para } k = -1 \\ 0 & \text{en otro caso} \end{cases}$$

Una secuencia Pseudo Aleatoria (PN) se genera por medio de un registro de corrimiento de n *Flip-flops* y un circuito lógico (sumador módulo 2), interconectados para formar un circuito de realimentación con múltiples lazos, como se muestra en la Fig. 1. .

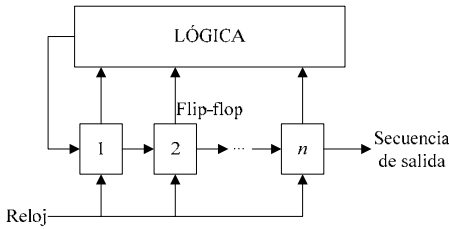


Fig. 1. Diagrama de bloques de un registro de corrimiento realimentado [7].

La secuencia de números Pseudos Aleatorios depende de la semilla (valores iniciales en los *Flip-flops*) y del polinomio generador (determina las conexiones de los *Flip-flops* con el bloque de lógica). La secuencia PN implementada fue de máxima longitud y valores de unos y menos unos.

La matriz Hadamard de 2×2 es:

$$\mathbf{M}_1 = \begin{bmatrix} -1 & -1 \\ -1 & 1 \end{bmatrix}$$

La matriz de Hadamard de $2^n \times 2^n$ puede ser construida recursivamente:

$$\mathbf{M}_n = \begin{bmatrix} \mathbf{M}_{n-1} & \mathbf{M}_{n-1} \\ \mathbf{M}_{n-1} & -\mathbf{M}_{n-1} \end{bmatrix}$$

La matriz Hadamard Aleatoria, se construyó a partir de la matriz Hadamard desordenando con distribución aleatoria uniforme sus filas.

La matriz Sinusoidal se construyó de la siguiente manera:

$$\varphi_i[n] = \cos(\omega_{i-1}n) \quad , \text{para } 1 \leq i \leq 128$$

$$\varphi_i[n] = \cos(\omega_{i-1}n + \pi/2) \quad , \text{para } 129 \leq i \leq 256$$

donde $\omega_i = (i\pi)/128$ y $n = 1, 2, \dots, N$.

VI. RESULTADOS Y ANÁLISIS

Las gráficas estadísticas realizadas para las matrices Bernoulli, Gaussiana y Pseudo Aleatoria arrojaron resultados muy similares, por consiguiente se analizan conjuntamente en esta Sección y se referirá de ahora en adelante a éstas, como matrices BGP.

A. Estadísticas del PR

En la Fig. 2. , se ilustra el PR para la matriz Hadamard Aleatoria. Esta matriz arrojó las curvas más cercanas al 100% y aquellas con mayor pendiente para el menor número de vectores de medición N .

Cuando el número de vectores de medición es pequeño el PR es cercano a cero. A medida que se incrementa N la curva del PR crece, presentando un punto de pendiente máxima en $N_{\max}[m]$, donde m es el nivel de dispersión de la señal; por ejemplo, $N_{\max}[4]=16$. A partir de $N_{\max}[m]$ la pendiente de la curva del PR comienza a disminuir. Se encontró que $N_{\max}[m]$ coincide con porcentajes de recuperación en el rango del 25% al 75%. Para cada matriz el número de vectores de medición es dividido en 3 zonas de acuerdo el desempeño del algoritmo:

- *Zona de destrucción*: para la cual el algoritmo en promedio presenta un PR menor al 25%.
- *Zona de transición*: para la cual el algoritmo en promedio presenta un PR entre el 25% y 75%.
- *Zona de recuperación*: para la cual el algoritmo en promedio presenta un PR superior al 75%.

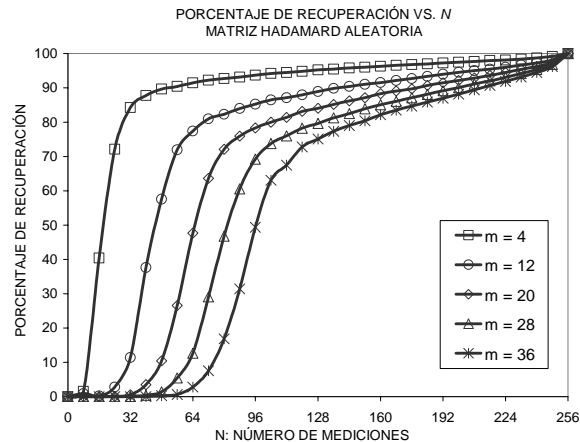


Fig. 2. Promedio muestral del PR vs. N para la matriz Hadamard Aleatoria.

Las matrices BGP logran un porcentaje de recuperación superior al 80%, para todos los niveles de dispersión, excepto para $m=36$, con el cual se alcanza un 78%, en el mejor de los casos.

En cuanto a la matriz Sinusoidal se tiene que para los niveles de dispersión de estudio, excepto para $m=4$, el algoritmo OMP requiere muchos vectores de medición para conseguir una buena recuperación de la señal.

En la Fig. 3. , se ilustran las gráficas de desviación estándar del PR. En la zona de destrucción se obtiene una mala recuperación para la mayoría de señales, por lo cual se presenta una baja desviación. En la zona de transición el promedio del PR mejora, sin embargo éste presenta una alta

varianza que alcanza su máximo en $N_{\max}[m]$, por lo que no se tiene certeza del desempeño del algoritmo. Para señales con pequeños valores de dispersión se presentan campanas de desviación angostas y con picos pronunciados, mientras que para señales con un nivel de dispersión más alto se obtienen resultados con una varianza menor, pero repartida sobre un conjunto más amplio de vectores de medición. La varianza en la zona de recuperación es una función monótonamente decreciente.

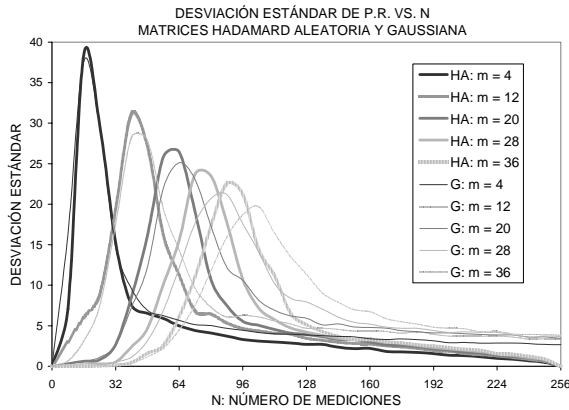


Fig. 3. Desviación estándar vs. N para la matrices Hadamard Aleatoria (HA) y Gaussiana (G).

En la Fig. 3., se observa que las curvas de la matriz Hadamard Aleatoria decaen más rápidamente que las matrices BGP. También se aprecia que las campanas de la matriz Hadamard Aleatoria son más angostas, lo que indica que la incertidumbre en la recuperación se concentra en un número menor de vectores de medición.

Para una mejor visualización de cómo se distribuyen los PR para los diferentes valores de N se ilustra en la Fig. 4. un histograma tridimensional de los resultados obtenidos. Se realizaron las siguientes observaciones:

- A medida que aumenta N las barras son más altas y la prominencia que forman se adelgaza. Lo que ilustra que la desviación estándar de los porcentajes de recuperación, disminuye con el aumento de N .
- Se puede observar que en la zona de transición las barras son extremadamente bajas y dispersas, lo que indica que en esta zona no se tiene certeza de la recuperación la señal.

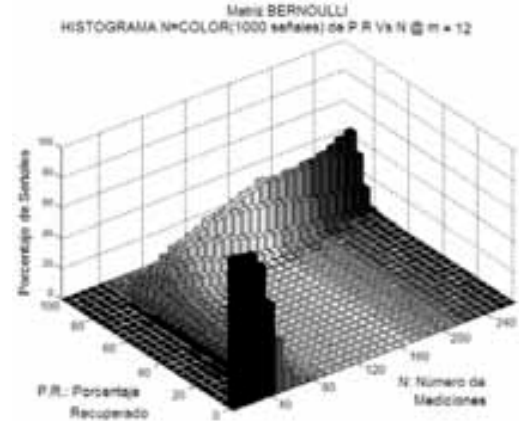


Fig. 4. Histograma tridimensional del PR para la matriz Bernoulli $m = 12$.

B. Obtención del número de vectores de medición

En esta Sección se comparan directamente las matrices de medición con base en los resultados obtenidos del algoritmo OMP. La Fig. 5. permite calcular el mínimo número de vectores de medición N requerido para alcanzar la zona de recuperación, con un nivel de dispersión dado, para cada una de las matrices de medición propuestas.

La mejor curva es aquella que utilice el menor N para obtener un PR igual a 75%. Naturalmente si se escoge un N mayor al indicado en la Fig. 5., el PR será mayor. Se observa que la matriz Hadamard Aleatoria supera a todas las demás, ya que su curva está por debajo de las otras para todos los niveles de dispersión.

Existen puntos de interés en las gráficas de la Fig. 5. que corresponden a las intersecciones de las diferentes curvas. Estos puntos establecen para qué intervalos en el nivel de dispersión una matriz de medición se desempeña mejor que otra. De esta manera, comparando las curvas de las matrices Sinusoidal, con las de las matrices BGP, se encontró que para valores de m entre 4 y 27 las matrices BGP son mejor opción, mientras que para el intervalo de m comprendido entre 27 y 36, la matriz Sinusoidal las supera. Análogamente se observa una situación similar con la matriz Hadamard en el punto de intersección $m = 28$.

Así como la matriz Hadamard Aleatoria superó las expectativas, también se encontró que la matriz Hadamard es la de menor PR, en el intervalo de dispersión entre 4 y 28.

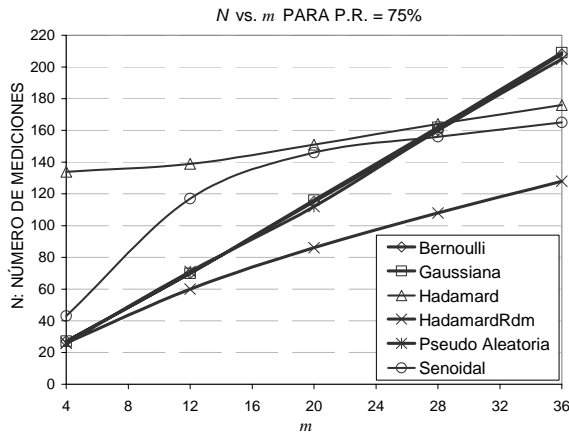


Fig. 5. Gráfica de número de mediciones vs. nivel de dispersión, para obtener un PR = 75%.

VII. TRABAJOS RELACIONADOS

En distintos trabajos se han introducido diversos arreglos de matrices y metodologías para evaluarlas. En este trabajo se presenta una comparación directa de las matrices, donde se emplearon herramientas de evaluación basadas en mediciones estadísticas de los resultados que arroja el algoritmo OMP al emplear cada una de las matrices.

En [6], los resultados de la evaluación de las matrices Benoulli y Gaussiana con el algoritmo OMP, muestran un comportamiento casi idéntico en la recuperación de la señal, lo cual está en concordancia con los resultados obtenidos en este trabajo. Las diferencias entre ambos trabajos radican en que en [6] se define el porcentaje de señales recuperadas a partir de la recuperación exacta de cada una de las señales, y en el caso de este trabajo se definió una medida (porcentaje de recuperación) que evalúa la recuperación a partir de la norma euclidiana del error.

Del conjunto de matrices evaluadas, la matriz Hadamard Aleatoria, propuesta en este trabajo, es aquella con la que se obtienen los mejores resultados al recuperar señales dispersas utilizando el algoritmo OMP.

VIII. CONCLUSIONES

Se encontró la matriz que a partir del menor número de vectores de medición, logró la recuperación más adecuada de la señal. Esta es la matriz Hadamard Aleatoria, la cual es la mejor opción dentro del conjunto de matrices estudiadas, para ser implementada con el algoritmo OMP.

Pese a que la matriz Pseudo Aleatoria no arrojó los mejores resultados, ésta presenta una ventaja muy importante: su fácil generación. A partir de una semilla inicial se puede construir siempre la misma secuencia de números aleatorios y por tanto reconstruir idénticamente la matriz. Esto último puede ser aplicado en sistemas de criptografía, en los cuales se encripta la información del mensaje por medio del vector de observación y en el receptor se descrypta el mensaje

utilizando el algoritmo OMP. Naturalmente los únicos que conocen la semilla y el polinomio generador son el sistema transmisor y el receptor, más no los escuchas que pueda tener el canal.

Se encontró que las matrices BGP presentan comportamientos casi idénticos al ser implementadas como matrices de medición en el algoritmo OMP. Excluyendo a la matriz Hadamard Aleatoria, se comparó el desempeño de las otras matrices y se encontró que existen dos intervalos en el nivel de dispersión para los cuales se tienen diferentes desempeños de las matrices. Analizando las curvas de las matrices Sinusoidal, Bernoulli, Gaussiana y Pseudo Aleatoria, se encontró que estas tres últimas para valores de m entre 4 y 27 son la mejor opción, mientras que para el intervalo de m comprendido entre 27 y 36, la matriz Sinusoidal las supera.

En la compresión de los datos utilizando Muestreo Compresivo, existe pérdida de la información que se ve reflejada en el porcentaje de error al recuperar la señal. Dependiendo de la aplicación se deben definir los niveles para una recuperación aceptable de los datos y a partir de estos determinar el nivel de compresión de los mismos.

IX. REFERENCIAS BIBLIOGRÁFICAS

- [1] R. Baraniuk, *Compressive Sensing [Lecture Notes]*, IEEE Signal Processing Magazine, Vol. 24, pp. 118-121, 2007.
- [2] C. La, M. Do, *Tree-based Orthogonal Matching Pursuit Algorithm for Signal Reconstruction*, in IEEE International Conference on Image Processing, pp. 1277-1280, 2006
- [3] M. Wakin, J. Laska, M. Duarte, D. Baron, S. Sarvotham, D. Takhar, K. Kelly, R. Baraniuk, *An Architecture for compressive imaging*, IEEE International Conference on Image Processing, pp. 1273-1276, 2006.
- [4] S. Kirolos, J. Laska, M. Wakin, M. Duarte, D. Baron, T. Ragheb, Y. Massoud R. Baraniuk, *Analog-to-Information Conversion via Random Demodulation*, disponible en www.ece.rice.edu/~jnl5066/papers/DCAS2006_randmod.pdf, 2007.
- [5] J. Tropp, *Greed is Good: Algorithmic Results for Sparse Approximation*, IEEE Trans. Inform. Theory, Vol. 50, pp. 2231-2241, 2004.
- [6] J. Tropp, A. Gilbert, *Signal Recovery from Random Measurements via Orthogonal Matching Pursuit*, disponible en: www.math.lsa.umich.edu/~annacg/papers/TG05-signal-recovery-rev-v2.pdf, 2005.
- [7] S. Haykin, *Sistemas de Comunicación*, México: Limusa Wiley, pp. 479-487, 2002.
- [8] E. Candes, *Compressive sampling*, in Proceedings of the International Congress of Mathematicians, Madrid, Spain, 2006.
- [9] S. Mallat, Z. Zhang, *Matching Pursuit with Time Frequency Dictionaries*, IEEE Transactions in Signal Processing, December 1993.

AUTORES



Adolfo León Recio Vélez, obtuvo su grado como Ingeniero Electrónico en la Pontificia Universidad Javeriana, Bogotá en 1993, y de maestría en The Ohio State University, Columbus, OH en 1999. Actualmente adelanta estudios de doctorado en Virginia Tech, Blacksburg, VA. Sus áreas de interés son Procesamiento Digital de Señales, Comunicaciones, Radio Cognitiva y Computación Reconfigurable.

adolfo.recio@javeriana.edu.co



Julián Armando Quiroga Sepúlveda, obtuvo su grado como Ingeniero Electrónico de la Universidad Nacional de Colombia, Manizales, en 2004 y de maestría en la Universidad de los Andes, Bogotá, en 2006. Actualmente se desempeña como profesor del Departamento de Electrónica de la Pontificia Universidad Javeriana, Bogotá. Sus áreas de interés incluyen el

Procesamiento Digital de Señales, Visión Artificial y Machine Learning. quiroga.j@javeriana.edu.co



Adriana Patricia Lobato Polo, recibió su título de Ingeniero Electrónico de la Pontificia Universidad Javeriana en 2008. Sus áreas de interés son Sistemas de Comunicaciones y Bioingeniería.

aplpolo@yahoo.com



Rafael Humberto Ruiz Coral, recibió su título de Ingeniero Electrónico de la Pontificia Universidad Javeriana en 2008. Sus áreas de interés son Sistemas de Comunicaciones y Procesamiento de Señales.

ruiz.rafa@yahoo.com

Pathfinder: Estimación de la trayectoria de un vehículo a partir de una cámara embarcada

Ricardo Andrés Calderón Capote, Juan Guillermo Perdomo Torres, Juan Pablo Martínez Rojas,
Director: Ing. Alejandro Forero Guzmán M.Sc.

r.calderon@javeriana.edu.co, perdomo-j@javeriana.edu.co, jp.martinez@javeriana.edu.co,
alejandro.forero@javeriana.edu.co

Pontificia Universidad Javeriana – Bogotá

Resumen— El presente proyecto desarrolla un sistema para estimar la trayectoria descrita por un vehículo, a partir de un video capturado en una cámara embarcada en el mismo vehículo. El video que se obtiene es analizado por medio de técnicas de estimación de flujo óptico; estas técnicas se implementan en algoritmos desarrollados con la librería OpenCV, para procesar y extraer la información relevante para estimar la trayectoria del vehículo.

Index Terms— Image processing; image sequence analysis; optical flow; video signal processing.

I. INTRODUCCIÓN

El sistema de transporte vehicular es esencial para el desarrollo y crecimiento de una nación. Constantemente en el mundo se están añadiendo nuevas técnicas a los sistemas de control de flujo de tráfico vehicular, basados en estudios relacionados con la descripción y la cuantificación del flujo vehicular [1].

El presente proyecto desarrolla un sistema para estimar la trayectoria descrita por un vehículo, a partir de un video capturado en una cámara embarcada en el mismo vehículo; el sistema arrojará información de la dirección del vehículo, la cual podría ser unificada a un sistema de control de flujo de tráfico vehicular para enriquecerlo con la perspectiva que se tiene desde el interior de los móviles que pretende controlar el sistema.

El video que se obtiene desde la cámara embarcada en el vehículo es analizado por medio de técnicas de estimación de flujo óptico; estas técnicas se implementan en algoritmos desarrollados con la librería OpenCV, para procesar y extraer la información relevante para estimar la trayectoria del vehículo.

Para estimar la trayectoria de un vehículo desde una cámara embarcada en él, es necesario comenzar con conceptos básicos de análisis de movimiento de un cuerpo en una imagen o video, para detectar los movimientos relativos entre la cámara y el entorno.

La estrategia planteada para detectar los movimientos relativos en una secuencia de imágenes será los algoritmos de flujo óptico ya que el único dato disponible en la secuencia de imágenes es la variación espacial y temporal del patrón de brillo de la imagen [4]. El campo de movimiento y el flujo óptico son iguales sólo en el caso de que las variaciones espaciales del patrón de brillo correspondan a características estructurales de las superficies [5].

Una parte fundamental de este proyecto es la creación de una base de datos de videos capturados desde una cámara embarcada en un vehículo en diferentes condiciones no controladas como lo son; tráfico vehicular, luminosidad del ambiente, en vías urbanas y rurales. Después se realiza el procesamiento de los videos capturados con la librería OpenCV, aprovechando que dicha librería está desarrollada para la optimización del procesamiento de imágenes y videos con la arquitectura de procesadores Intel [6]. Otra ventaja de la librería OpenCV es la existencia de funciones especializadas para el análisis de movimiento en imágenes con las técnicas para la estimación de flujo óptico de Lucas y Kanade, Horn y Schunck y la de igualación de bloques [5].

La información obtenida aplicando las técnicas de flujo óptico es analizada para poder identificar los patrones relevantes que puedan ser útiles para el desarrollo de un algoritmo que estime la dirección de un vehículo y así lograr describir la dirección del movimiento instantáneo del vehículo a lo largo de un intervalo de tiempo.

A continuación en las secciones II y III se explica la metodología utilizada para el desarrollo de la problemática planteada en este documento, con un análisis de los resultados

obtenidos y finalmente las conclusiones respectivas del proceso realizado.

II. DESCRIPCIÓN

A. Procesamiento de video e imágenes.

En la primera fase de desarrollo se convierte el video capturado en una secuencia de imágenes (frames). Cada imagen que compone la secuencia de imágenes es procesada para obtener la misma imagen en escala de grises y otra en RGB, mediante la función `cvCreateImage` que se encuentra en la librería OpenCV, debido a que la función de estimación de flujo óptico en la librería OpenCV requiere como parámetro de entrada que la imagen este en escala de grises.

B. Estimación del flujo óptico.

El flujo óptico mide el movimiento de cada pixel a partir de la comparación entre el cuadro de video actual y el cuadro de video siguiente, ubicando la variación espacial y temporal del patrón de brillo de la imagen. [5]

Los algoritmos de interpretación de secuencia de imágenes que requieren la extracción del flujo óptico, están basados en dos métodos, el método de los gradientes y el método de Segmentación. El método de los gradientes de espacio-temporales consiste en determinar los cambios (gradientes) espaciales y temporales del patrón de grises de la imagen y a partir de ellos obtener el flujo óptico. [5] y [7].

El desarrollo de este proyecto está basado en el uso de la técnica de estimación de flujo óptico de Lucas y Kanade de pirámides que hace uso de la ecuación de restricción de movimiento que relaciona el flujo óptico (u,v) con los gradientes o variaciones espacio-temporales (E_x, E_y, E_t)

$$E_x u + E_y v + E_t = 0 \quad (1) [7]$$

Esta ecuación contiene dos incógnitas (u, v) , por lo que no es posible obtener una única solución a partir de ella. No puede ser determinada la componente del movimiento perpendicular al gradiente del brillo, y tampoco puede determinarse en los casos de zonas de igual brillo (iso-brillo). Esto se conoce con el nombre de problema de apertura. [7]

Otra condición para la determinación del flujo óptico es que los cambios en el movimiento entre puntos cercanos de la imagen deben ser suaves lo que lleva al desarrollo de las siguientes ecuaciones iterativas: [5]

$$u = \hat{u} - \frac{E_x(E_x \hat{u} + E_y \hat{v} + E_t)}{E_x^2 + E_y^2} \quad (2)$$

$$v = \hat{v} - \frac{E_y(E_x \hat{u} + E_y \hat{v} + E_t)}{E_x^2 + E_y^2} \quad (3)$$

El uso de la técnica de Lucas y Kanade para la estimación de la trayectoria de un vehículo, arroja más información relevante para la estimación de flujo óptico respecto a las otras

funciones de flujo óptico disponibles en la librería de OpenCV, ya que proporciona más parámetros de salida que son relevantes para la estimación de la trayectoria del vehículo.

La técnica de Lucas y Kanade para la estimación de flujo óptico consiste en utilizar un método de multiresolución o pirámides que compara los pixeles del frame actual con el siguiente mediante una ventana que aumenta su área de comparación cada vez que realiza una iteración hasta que la ventana llega a tener la misma área que el cuadro de imagen. (Ver Figura 1.)

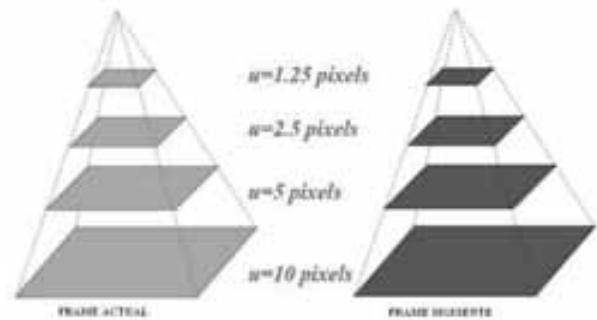


Figura 1. Técnica de multiresolución.

Los datos arrojados por la función de flujo óptico de Lucas y Kanade de la librería OpenCV, son analizados y procesados mediante un algoritmo que determina el desplazamiento del pixel encontrado mediante la relación de la posición escalar del pixel en el frame actual con la posición escalar del mismo pixel en el frame siguiente, arrojan dos tipos de información, magnitud y ángulo del flujo óptico en un plano coordenado

C. Determinación de la trayectoria.

El procesamiento de la secuencia de imágenes comienza con la limitación de unas regiones de la imagen que contiene la mayor información para la estimación del flujo óptico; las regiones de interés para esta aplicación son los sectores izquierdo y derecho de la imagen ya que las regiones donde el movimiento de pixeles es mayor respecto a la secuencia de imágenes, las regiones creadas son procesadas con la técnica de Lucas y Kanade para estimar el flujo óptico y generalizar estos datos a todo el cuadro de imagen. (Ver Figura 2.)

La información de magnitud y ángulo del flujo óptico de las regiones de interés, es analizada por otro algoritmo que realiza un ordenamiento de los datos para clasificarlos usando la teoría de probabilidad de histograma, para luego ser correlacionados entre si y lograr estimar la dirección de movimiento del vehículo en un intervalo de tiempo.



a) Imagen completa



b) Región 1

c) Región 2

Figura 2. Regiones de interés de la imagen procesadas

III. ANÁLISIS DE RESULTADOS

Para realizar un análisis del sistema, se tomaron una serie de videos desde el interior del vehículo con la cámara ubicada en centro del parabrisas, en condiciones no controladas de ambiente, con el fin de observar la efectividad del programa y la capacidad de poder estimar la dirección de la trayectoria del vehículo.

Para bajar la carga computacional, y teniendo presente la velocidad del vehículo, se decide analizar el video en tripletas de frames, los resultados obtenidos se pueden observar en la tabla 1. El promedio de efectividad fue calculado por el número de tripletas de frames acertadas dentro del total de tripletas de cada video. Las tripletas de frames de cada video donde su resultado fue erróneo tienen múltiples problemas, mayormente generados por la presencia de brillo constante en la imagen, presencia de flujo vehicular y comparación erróneas de pixeles.

ESCALA DE EFECTIVIDAD (%)	NUMERO DE VIDEOS	PROMEDIO EFECTIVIDAD (%)
100- 95	12	96.3
94-90	5	92.7
89-85	7	88.2
84-80	4	82.3
84-00	2	78.4
Total	30	90.7

Tabla 1. Efectividad del sistema.



a) Triplet 1

b) Triplet 2

Figura 3. Brillo constante en las imágenes

En la Figura 3. se puede observar que de una tripletta de frames a otra no hay flujo óptico en el pasto de color verde, debido a que presenta un brillo constante lo que genera un error en la dirección de flujo óptico.



Figura 4. Imagen con flujo vehicular.

En la figura 4. Las líneas amarillas representan el desplazamiento de los pixeles de frame a frame. Se observa la presencia de un vehículo en la imagen; que afecta la información necesaria para estimar la trayectoria del vehículo que tiene la cámara embarcada, debido a que genera líneas de

flujo de sentido contrario al verdadero movimiento relativo del vehículo que tiene la cámara embarcada.

El problema de comparación errónea de píxeles se encuentra cuando la secuencia de imágenes contiene características de zonas pequeñas iso brillo, arrojando una comparación de un pixel con otro pixel distinto que tiene el mismo brillo. Generando flujo óptico con magnitudes y ángulos incoherentes. (Ver Figura 4.)



Figura 4. Imagen con magnitudes y ángulos erróneos.

IV CONCLUSIONES

Las pruebas realizadas al sistema mostraron un buen resultado frente a los factores que pueden afectar la estimación de la dirección del vehículo como lo son: las condiciones ambientales no controlables y la no limitación del sistema a condiciones que no se asemejan a la realidad del tráfico vehicular en la cual se desea implementar este sistema.

Existen diferentes técnicas para la estimación de flujo óptico como se explican en este documento, la técnica de Lukas y Kanade utilizada en el presente programa tiene la ventaja de proporcionar la información necesaria para poder estimar la dirección de la trayectoria del vehículo demostrado en los resultados obtenidos. Sin embargo en ambientes donde hay zonas con mayor contraste la técnica de la estimación de flujo óptico basada en el gradiente de brillos de la imagen tiene mayor efectividad.

La librería de OpenCv en comparación con programas como MATLAB, realiza el procesamiento de los videos e imágenes de una manera más rápida ofreciendo un mejor desempeño para la futura implementación de este programa.

La generalización de la información necesaria para la estimación de la trayectoria del vehículo a partir de regiones de interés ayudando a eliminar la información no deseada y el ruido presente en la imagen, logrado un mayor desempeño computacional.

El uso de la teoría de probabilidad para la manipulación de la información en los algoritmos, hace que los datos sean organizados y procesados de una manera más adecuada para hallar la magnitud y el ángulo del flujo óptico.

El trabajo realizado en este proyecto representa el inicio de un proyecto macro en esta área liderado por la sección de comunicaciones del departamento de electrónica de la Pontificia Universidad Javeriana, ya que como se dijo anteriormente la motivación de este trabajo es poder implementar el programa unificado con otros proyectos enfocados en la problemática de control de flujo vehicular, por ello se quiere que este trabajo sea la base para futuras profundizaciones en este aspecto y mejorar los parámetros que limitan por ahora a esta investigación.

BIBLIOGRAFÍA

- [1] N. Garber y L. Hoel, Ingeniería de tránsito y carreteras, 3ª Ed., México D.F.: Thompson Editores S.A., 2005, pp 1.
- [2] N. Garber y L. Hoel, "Principios Fundamentales del flujo de tránsito," Ingeniería de tránsito y carreteras, 3ª Ed., México D.F.: Thompson Editores S.A., 2005, pp 175.
- [3] N. Garber y L. Hoel, "Estudios de ingeniería de tránsito," Ingeniería de tránsito y carreteras, 3ª Ed., México D.F.: Thompson Editores S.A., 2005, pp 77.
- [4] A. Zuloaga, J. Martín, L. Nozal, "Determinación del movimiento a partir de secuencias de imágenes - Procesamiento primario," Mundo Electrónico, n. 278, pp. 55-59, Junio 1997.
- [5] A. Verri y T. Poggio. "Motion field and optical flow: qualitative properties". IEEE Transactions on pattern analysis and machine intelligence, vol. 11, no. 5, pp. 490-498, May. 1989.
- [6] Intel Corporation, Open Source Computer Vision Library, Intel Corporation, 2000.
- [7] B. Horn y B. Schunck, "Determining Optical Flow". Artificial Intelligence, vol 17, pp 187-203, 1981

Estimación del fondo para la detección de vehículos en video

Julián Quiroga Sepúlveda, Juan Manuel Quintero Angarita, Pablo Andrés Quintero Armero,
María Ximena Weiss Maldonado

quiroga.j@javeriana.edu.co, juanquintero@javeriana.edu.co, quinterop@javeriana.edu.co,
mweiss@javeriana.edu.co

Pontificia Universidad Javeriana - Bogotá

Resumen—En este trabajo se presenta un algoritmo para la estimación del fondo de una escena a partir de un conjunto de imágenes. La información de la varianza temporal de cada píxel es utilizada para determinar cuales de ellos pertenecen al fondo y cuales a un objeto ajeno a la escena. A partir de esta información, se computa un conjunto de fondos parciales, utilizando ventanas temporales no traslapadas del video de la escena. El fondo es actualizado periódicamente con el fin de lidiar con los posibles cambios de iluminación. El fondo de la escena es estimado combinando los fondos parciales por medio del filtro mediana. Finalmente se presentan algunos resultados y conclusiones sobre el algoritmo desarrollado.

Palabras Claves—Varianza, filtro mediana, erosión, dilatación.

I. INTRODUCCION

EL creciente número de automóviles en las ciudades junto a las falencias que presentan las vías en nuestro país, hacen que transitar por las ciudades sea una tarea difícil y bastante estresante. La movilidad del parque automotor se ve altamente afectada por los constantes estancamientos vehiculares generados por la imprudencia de los conductores, el mal diseño de las vías y el exceso de vehículos. Por tal motivo, cualquier tipo de información que permita determinar las causas que generan dichas alteraciones en el flujo vehicular podría ser útil a la hora de generar alternativas para mejorar la movilidad. Las principales variables de tráfico son la cantidad de vehículos y su velocidad. Existen otras variables que pueden determinarse a partir de estas como el flujo vehicular y otras independientes, que pueden ser también de utilidad, como la clase de vehículo.

Por lo general, la adquisición de las variables de tráfico se lleva a cabo de forma manual. Debido a la cantidad de información que se presenta en un vía con alto flujo vehicular, este enfoque es poco apropiado, ya que pueden presentarse errores en la medición, pérdida de información y algunas variables son difícilmente medidas por simple inspección visual. Por tal motivo, actualmente en el mundo se hacen grandes esfuerzos para la adquisición automática de dichas

variables de tráfico utilizando diferentes tecnologías, entre las cuales se encuentra la visión artificial, área que contiene este trabajo. Sin embargo, debido a la complejidad del problema que plantea la adquisición automática de estas variables, la obtención de un sistema robusto ante cualquier ambiente es una tarea difícil. Diferentes características propias de cada vía, tales como su diseño, la densidad y velocidad vehicular, el tipo de vehículos, el comportamiento de los conductores, entre otros, repercuten en forma directa en el diseño del sistema de adquisición. Así mismo, características propias del sistema, como número de cámaras y su ubicación también afectan el desempeño del sistema.

La principal tarea en un sistema de adquisición automático consiste en la detección de los vehículos presentes en la escena. Las estrategias más comunes para la detección pueden dividirse en dos; las basadas en movimiento y las basadas en la sustracción del fondo; en muchos casos, las dos estrategias son utilizadas de forma conjunta para obtener un sistema más robusto. En este trabajo se presenta un método para la estimación del fondo a partir de un conjunto de imágenes de una escena, como primer paso para la detección de los vehículos presentes en la misma.

La estimación del fondo permite identificar objetos ajenos a la escena, siempre y cuando éstos presenten un contraste suficiente; caso en el cual la detección se puede realizar mediante la sustracción del fondo a una imagen determinada. La calidad del fondo estimado es altamente dependiente de las características del tráfico tales como la cantidad, velocidad, tamaño y color de los vehículos, y de las condiciones de adquisición, como la posición de la cámara. Adicionalmente, las características del fondo a estimar pueden variar con el tiempo debido a cambios en la iluminación, por lo que se requiere un algoritmo adaptable o una estimación del fondo cada determinado tiempo.

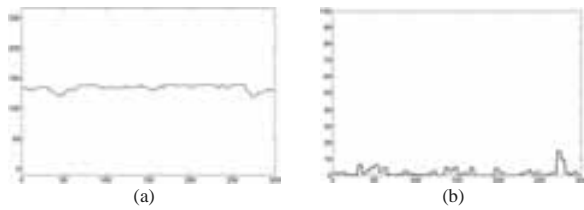


Figura 1. Comportamiento del un píxel del fondo. a) Valor en el tiempo y b) Varianza temporal.

En la literatura la mayoría de autores han optado por la utilización de algoritmos adaptativos para la estimación, debido a que no necesitan ser inicializados y pueden ser más robustos a cambios fuertes en la iluminación [1]. Stauffer et al [2] y [3], modelan el valor de cada píxel del fondo en el tiempo como una combinación de funciones gaussianas. Sobre una pequeña ventana de tiempo determinan la función gaussiana más similar a los valores presentados por el píxel de fondo en ese intervalo de tiempo. Utilizando esta similitud se determina si el píxel pertenece o no al fondo y la forma en que deben actualizarse las demás funciones gaussianas. Gao et al [4] proponen un modelo basado en el filtro Kalman para estimar cada píxel del fondo. Los parámetros del filtro son ajustados utilizando un filtro adaptativo RLS. En este trabajo se presenta un algoritmo adaptativo basado en un principio más simple, que pretende brindar una solución menos costosa computacionalmente y más enfocada hacia la detección de vehículos en una vía determinada.

El presente trabajo está dividido como sigue. En la Sección II se presenta la idea general del algoritmo propuesto. En la Sección III se describe la implementación del algoritmo para la estimación de fondo. En la Sección IV se presentan algunos de los resultados obtenidos. Finalmente, en la Sección V se concluye sobre el trabajo realizado.

II. ESTIMACION DEL FONDO

A. Información temporal de cada píxel

La estimación del fondo parte del hecho de que el valor de un píxel perteneciente al fondo debe permanecer relativamente constante a través del tiempo. Sin embargo, existen diferentes factores que dificultan la verificación de esta simple restricción. El primero de ellos es el ruido presente en el video. Al observar el valor de un píxel del fondo en el tiempo, bajo una iluminación constante, se observan leves variaciones, como se ilustra en la Figura 1.a. Cada uno de los píxeles del fondo puede modelarse como un proceso estocástico gaussiano, cuya varianza se encuentra dentro de un rango bajo de valores. En la Figura 1.b se ilustra la varianza del píxel del fondo, la cual es estimada utilizando el valor del píxel sobre una ventana temporal de tres cuadros del video.

Sin embargo, cuando un objeto entra a la escena los píxeles de la imagen que lo contienen presentan variaciones en su valor, las cuales son proporcionales al contraste entre el fondo y el objeto. En la Figura 2.a se ilustra el valor de un píxel del

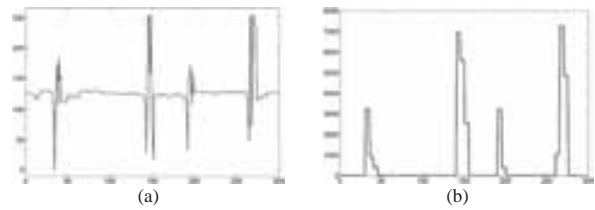


Figura 2. Varianza temporal de un píxel del fondo sobre el cual se presentan varios vehículos: a) valor en el tiempo y b) varianza computada.

fondo en el tiempo, sobre el cual se presentan un conjunto de vehículos cada determinado tiempo. Se puede observar que la presencia de los objetos es fácilmente identificada sobre el valor de píxel, presentado valores mayores para vehículos más claros debido al color gris de la vía. Si es computada la varianza temporal del valor del píxel, como se ilustra en la Figura 2.b, puede determinarse en que instantes del tiempo se encuentra un objeto ajeno a la escena presente en la misma.

El algoritmo propuesto en este trabajo utiliza la información de la varianza temporal de los píxeles de la escena con el fin de determinar cuales de ellos pertenecen al fondo y cuales otros pertenecen a un objeto en movimiento. Una vez que un píxel es etiquetado como parte del fondo, éste es utilizado para actualizar el fondo de la escena. Por otro lado, si un píxel es etiquetado como parte un objeto móvil, dicha posición no es actualizada y el valor anterior del fondo es conservado.

B. Máscara de actualización

Con el fin de estimar adaptativamente el fondo de la escena, se debe determinar cuales píxeles deben ser actualizados. Sin embargo, dado que los objetos móviles corresponden a vehículos, cuyo tamaño es representativo en la imagen, debe determinarse cuales son las regiones que deben ser actualizadas. Para tal fin, primero son encontrados los píxeles que pertenecen al fondo y los píxeles que pertenecen a los objetos móviles. La varianza temporal de cada píxel es computada sobre cada componente de color y el máximo entre esta tres es utilizado para conformar un imagen única de varianza. Cada píxel de la imagen es etiquetado como perteneciente al fondo o a un objeto de acuerdo con un umbral seleccionado de forma apropiada.

La selección de este umbral juega un papel fundamental en el desempeño del algoritmo. Un umbral bajo, es más sensible a pequeñas variaciones, detectando cambios de intensidad más pequeños, lo que es bueno para móviles que presenta bajo contraste. Sin embargo, puede llegar a ser muy sensible al ruido presente en el video, detectando así falsos objetos en movimiento. Por otro lado, si el umbral es muy alto, el algoritmo es más robusto al ruido pero algunos objetos en movimiento pueden perderse. Por tal motivo existe un compromiso entre la detección del verdadero movimiento y el ruido en el video. El umbral apropiado puede ser calculado aprovechando el hecho de que la mayoría de los píxeles de la imagen pertenecen al fondo, por lo cual deben presentar una baja varianza. Computando la moda de la varianza temporal

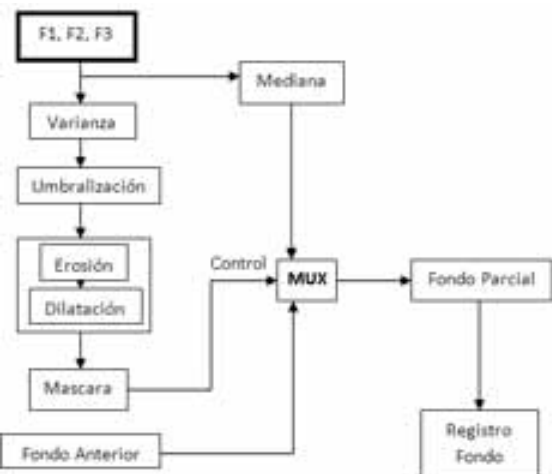


Figura 3. Diagrama en bloques del algoritmo.

de los píxeles de la imagen y asignado el umbral como 5 veces dicho valor se obtienen buenos resultados. Sin embargo, una selección manual del umbral generó resultados muy similares con un menor tiempo de cómputo.

Una vez se han determinado los píxeles del fondo y de los objetos, deben identificarse las regiones de la imagen que deben ser actualizadas. Para tal fin, en primera instancia son utilizados un proceso de erosión y otro de dilatación, en ese orden, con el fin de eliminar ruido y rellenar los objetos.

En este punto se tiene una máscara que indica los píxeles del fondo anterior que pueden ser actualizados y los píxeles que deben ser conservados.

C. Actualización del fondo

La actualización del fondo de la escena se realiza combinando un conjunto de fondos parciales. Cada fondo parcial es estimado utilizando un conjunto de n cuadros del video, por medio de la máscara de actualización computada a través de la varianza temporal de cada píxel. Un píxel de la máscara de valor 0 indica un píxel del fondo parcial que no debe ser actualizado, en este caso se conserva en dicha posición el valor del fondo actual. Un píxel de la máscara de valor 1 representa un píxel del fondo parcial que puede ser actualizado. En este caso la mediana entre los n píxeles de la ventana temporal es utilizada para actualizar el fondo parcial. Se selecciona la mediana en vez de su promedio con el fin de evitar valores extremos que puedan presentarse. Los fondos parciales no comparten cuadros del video entre sí.

Una vez se tienen un conjunto de N fondos parciales éstos son utilizados para realizar la estimación del fondo actual. Nuevamente, con el fin de evitar valores extremos que puedan afectar la estimación, se utiliza la mediana entre el conjunto N de fondos parciales. Cada vez que un nuevo fondo parcial es estimado, es decir cada n cuadros, éste entra a formar parte del conjunto de N fondos parciales utilizados para realizar una estimación más precisa del fondo actual, eliminando a su vez el fondo parcial más antiguo. Dado que los píxeles del fondo anterior son utilizados como valores para los píxeles que no se actualizan del próximo fondo parcial, el algoritmo

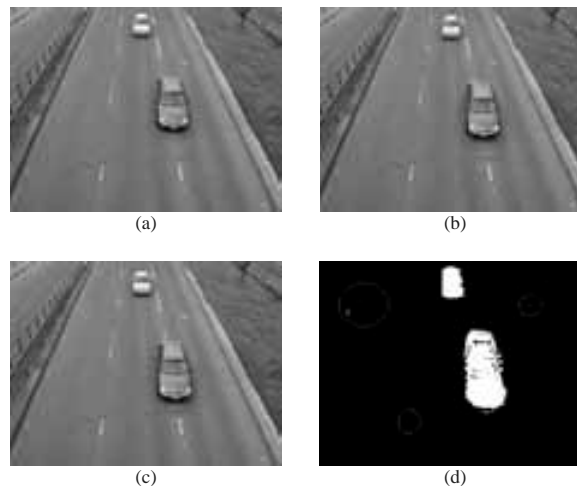


Figura 4. Cálculo de la varianza temporal: a), b) y c) cuadros consecutivos del video, d) imagen de varianza umbralizada.

cuenta con una memoria que es inversamente proporcional a la varianza temporal de los píxeles del fondo.

En la siguiente Sección se describe el algoritmo implementado para la estimación de fondo. En la Figura 3 se presentan las etapas principales del algoritmo.

III. ALGORITMO DE ESTIMACIÓN

Sea $I(x, y, n)$ el color del píxel (x, y) del cuadro n del video de la escena y sean I_R , I_G e I_B las componentes de rojo, verde y azul, respectivamente. Cada fondo parcial $f(x, y, k)$ es estimado utilizando tres cuadros consecutivos del video, como se ilustra en la Figura 4. La varianza temporal de cada píxel para cada componente de color, es estimada de acuerdo con

$$V_c(x, y, k) = \sum_{i=0}^2 [I_c(x, y, 3k-i) - \mu_c(x, y, k)]^2 \quad (1)$$

donde $C \in \{R, G, B\}$ y la media temporal μ está definida como

$$\mu_c(x, y, k) = \frac{1}{3} \sum_{i=0}^2 I_c(x, y, 3k-i) \quad (2)$$

El máximo valor sobre cada componente es calculado con el fin de conservar la variación temporal más alta, obteniendo así la imagen de varianza máxima V , definida como

$$V = \max\{V_R, V_G, V_B\} \quad (3)$$

donde la función máximo (**max**) es aplicada píxel a píxel.

La imagen de varianza se binariza utilizando el umbral definido en la Sección II.B. En las Figuras 4.a, 4.b y 4.c, se ilustran 3 cuadros consecutivos de un video, en los cuales un



Figura 5. Obtención de la máscara de actualización: a) resultado del proceso de erosión y b) resultado del proceso de dilatación.

par de vehículos se encuentran en movimiento. La Figura 4.d ilustra la imagen de varianza binarizada obtenida.

Con el fin de eliminar las partículas aisladas detectadas erróneamente como objetos móviles, como se puede ver en la Figura 4.d en círculos rojos, se realiza un proceso de erosión a la imagen de varianza binaria. La imagen obtenida después de la erosión se visualiza en la Figura 5.a, sobre la cual pueden observarse espacios vacíos presentes en cada uno de los objetos en movimiento. Por tal motivo, se realiza un proceso de dilatación a la imagen con el fin de rellenar dichos espacios y recuperar de forma aproximada la forma original del vehículo. Para tal fin, se utiliza un elemento estructural cuya geometría es análoga a la de un vehículo. Para este video el elemento es más alto que ancho debido a la orientación de la vía. En la Figura 5.b puede observarse el resultado.

La imagen obtenida después de aplicar los procesos de erosión y dilatación, es invertida con el fin de construir la máscara $M(x, y, k)$, en la cual los píxeles correspondientes a los objetos en movimiento tienen un valor cero, mientras los píxeles pertenecientes al fondo un valor uno.

El k -ésimo fondo parcial $f(x, y, k)$ es estimado utilizando la máscara obtenida, de acuerdo a:

- Si $M(x, y, k) = 0$ entonces $f(x, y, k) = F(x, y, k - 1)$, donde $F(x, y, k - 1)$ es el fondo anterior.
- Si $M(x, y, k) = 1$ entonces

$$f(x, y, k) = \text{med}[I(x, y, k), I(x, y, k + 1), I(x, y, k + 2)].$$

donde **med** implementa la función mediana.

Cada fondo parcial $f(x, y, k)$ es almacenado en un registro de fondos, con el fin de estimar el fondo de la escena. Para tal fin, se computa la mediana temporal de los últimos 5 fondos parciales, sobre cada componente de color. De esta forma se obtiene el fondo actual de la escena, el cual es utilizado para la estimación del siguiente fondo parcial y puede ser utilizado para la detección de vehículos.

IV. RESULTADOS OBTENIDOS

El algoritmo de estimación de fondo fue probado utilizando videos capturados desde un puente peatonal, sobre una vía de tres carriles de la ciudad de Bogotá.

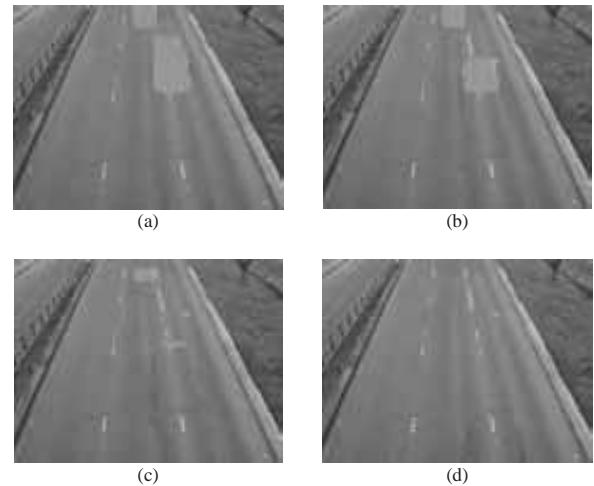


Figura 6. Proceso de estimación del fondo.



Figura 7. Sustracción del fondo estimado a un cuadro del video.

En la Figura 6, se ilustra el proceso de estimación para los primeros 21 cuadros de un video. En la Figura 6.a), se observa el primer fondo estimado, en donde pueden apreciarse valores incoherentes en la estimación (líneas verticales), debido a que no se cuenta con un fondo anterior. En las estimaciones posteriores el fondo estimado converge al fondo de la escena, como se ilustra en las Figuras 6.b)-d).

Una buena estimación del fondo permite realizar de forma sencilla y eficaz la detección de objetos ajenos a la escena. En las Figuras 7.a y 7.b se ilustra el resultado de sustraer el fondo estimado a un cuadro del video. Los objetos de interés pueden ser segmentados utilizando un umbral sencillo y procedimientos morfológicos sobre este resultado.

El algoritmo propuesto presenta un buen desempeño para vehículos pequeños en movimiento con suficiente contraste con respecto a la vía. Sin embargo, se encuentran ciertas fallas en el algoritmo original presentado para el caso de vehículos que contiene zonas de color uniforme, como los presentados en las Figuras 8.a y 8.b. A pesar del movimiento real de los vehículos, el movimiento aparente (el generado por las intensidades de los píxeles) es poco, pues en varios cuadros consecutivos del video se observa el mismo color. Por tal motivo, la varianza temporal es baja y de forma errónea dichos píxeles pertenecientes a los vehículos son asignados como parte del fondo, como se ilustra en las Figuras 8.c y 8.d. Con el fin de solucionar este problema, se utiliza la información de la máscara de actualización anterior para corregir la nueva máscara. Un cambio temporal en la máscara

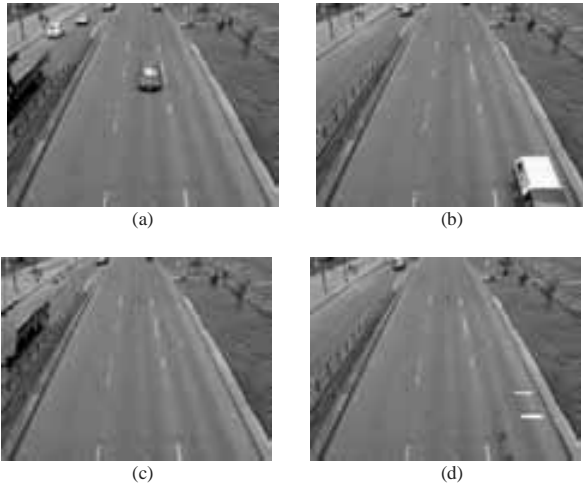


Figura 8. Error en la estimación debido a zonas de color uniforme.

de actualización de 0 a 1 (de objeto móvil a fondo) es aceptado sólo si se presenta un cambio suficiente en el color del píxel.

El desempeño del algoritmo es superior cuando la velocidad de los vehículos es alta y cuando el tiempo sobre el cual el fondo es visible es prolongado. Cuando un vehículo se detiene en la vía por un tiempo suficiente, este empieza a formar parte del fondo. Por tal motivo, es necesario tener información proveniente del sistema de detección de vehículos para evitar que un vehículo que se detiene en la vía forme parte del fondo estimado.

V. CONCLUSIONES

La varianza temporal del valor de los píxeles permite discriminar entre los píxeles que pertenecen al fondo y los que hacen parte de un objeto en movimiento. Para tal fin se requiere que el fondo y los objetos ajenos a él presenten un contraste suficiente. Dado al color gris de la vía, los vehículos claros o de colores fuertes presentan valores de varianza más altos a los de los carros oscuros cuando están en movimiento.

Por tal motivo, el valor del umbral de la varianza debe ser elegido de forma apropiada. Un valor bajo del umbral permite detectar vehículos en un mayor rango de color, sin embargo, hace al sistema más sensible al ruido. La selección automática del umbral en función de la moda de la varianza generó buenos resultados. Para esta selección se requiere que la mayor parte de los píxeles de todo cuadro del video sean parte del fondo.

La tasa de convergencia del algoritmo es superior cuando la velocidad de los vehículos es alta y se presenta un número bajo de vehículos. Se propuso una estrategia para lidiar con vehículos grandes con zonas de color uniforme, por medio de la información contenida en las máscaras anteriores de actualización.

Cuando la densidad vehicular es muy alta, el tiempo dentro del cual puede observarse el fondo es bastante corto, motivo por el cual, el algoritmo toma una gran cantidad de tiempo para computar una buena estimación del fondo. El algoritmo

necesita de información extrema para evitar que vehículos que se detienen en la vía por un periodo de tiempo lleguen a hacer parte del fondo.

La función mediana es utilizada para la estimación de los fondos parciales y para la estimación del fondo total. La mediana permite evitar valores extremos en la estimación de los fondos parciales y dota al algoritmo de estimación de fondo de una memoria inversamente proporcional a la varianza temporal presentada.

El presente algoritmo ha sido utilizado con éxito para la estimación del fondo sobre vías rápidas, en las cuales los vehículos no se detienen.

REFERENCIAS

- [1] Liyuan Li, Weimin Huang, Irene Yu-Hua Gu and Qi Tian, "Statistical Modeling of Complex Backgrounds for Foreground Object Detection," *IEEE Transactions on Image Processing*, vol. 13, pp.1459- 472, Nov. 2004.
- [2] C. Stauffer and W. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, pp. 747-757, Aug. 2000.
- [3] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1999.
- [4] Dashan Gao and Jie Zhou, "Adaptive Background Estimation for Real-time Traffic Monitoring," *IEEE Intelligent Transportation Systems Conference Proceedings.*, Aug. 2001.

Metodología para evaluar el difeomorfismo de un atractor caótico usando el filtro de Kalman

C. Ospina, L. D. Avendaño, E. Delgado, G. Castellanos

Resumen—Para la caracterización de señales fisiológicas, las cuales contienen estructuras fuertemente no lineales, es común el uso de procedimientos derivados de técnicas fractales que hacen parte del análisis de complejidad. En este trabajo se propone una función de evaluación basada en el filtro de Kalman para predecir puntos de un atractor reconstruido en el espacio de estados y medir la capacidad que tienen esos puntos para reconstruir la señal en el tiempo, y así, evaluar la calidad de ese atractor a partir de una señal unidimensional. Se propone el uso de medidas estadísticas como son la de Kullback–Leibler, Kolmogorov–Smirnov y Hellinger para determinar la diferencia entre la estructura estadística embebida en los puntos que fueron resultado de la predicción usando el atractor reconstruido y los puntos correspondientes de la señal original. Los resultados fueron obtenidos sobre la reconstrucción de atractores provenientes de señales ECG usando la base de datos del MIT-BIH y señales EEG obtenidas en la Clínica para Epilepsia de la Universidad de Bonn. Así, fue posible evaluar la capacidad de predicción correspondiente a los atractores reconstruidos a partir de los registros, con lo que se puede concluir que un atractor con alta capacidad de predicción en el tiempo implica buenas propiedades de embebimiento en el espacio de estados.

Palabras Claves—Embebimiento, dinámica no lineal, atractores y filtro de Kalman.

I. INTRODUCCIÓN

El análisis lineal atribuye el comportamiento irregular de un sistema a la naturaleza aleatoria de la señal de entrada, sin embargo, las entradas aleatorias no son las únicas fuentes posibles de irregularidad, ya que pueden atribuirse a estructuras no lineales impuestas en el modelo dinámico. Los métodos de dinámica no lineal que usualmente se utilizan para caracterizar señales fisiológicas basan su estudio en el atractor reconstruido en el espacio de estados a partir de la serie de tiempo que representa la dinámica del sistema biológico. La reconstrucción del espacio de estados se basa en el teorema de embebimiento [1], el cual establece que dadas las mediciones muestradas de la señal proveniente de un sistema, se puede reconstruir un espacio de estados difeomórfico al espacio de estados original, aunque desconocido, compuesto por todas las variables dinámicas.

En [2] se presenta un trabajo sobre registros magnetocardiográficos (MCG), donde se expone una metodología basada en la ley de embebimiento de Takens [3] y el método de reconstrucción de espacio de fase propuesto por Packard [4],

Carolina Ospina Aguirre, Luis David Avendaño Valencia, Edilson Delgado Trejos y Germán Castellanos, pertenecen al Grupo de Control y Procesamiento Digital de Señales, Universidad Nacional de Colombia, Departamento de Eléctrica, Electrónica y Computación, Vía al Magdalena, Km 7, La Enea, Campus la Nubia, Manizales, Caldas, Colombia. E-mail: {cospinaa, ldavendanov, edelgadot, cgcastellanosd}@unal.edu.co.

con el fin de obtener el atractor caótico correspondiente a la fuerza de inducción magnética cardíaca. En [5] se presenta un algoritmo para sincronizar dos sistemas caóticos diferentes, usando una combinación entre el filtro extendido de Kalman y técnicas de control en modo deslizante. Los resultados de simulación muestran el buen comportamiento del algoritmo en la sincronización de sistemas caóticos en presencia de ambientes ruidosos. En [6] se presenta un método iterativo llamado *asimilación de datos* para resolver el problema de estimar estados de un sistema dinámico usando tanto las observaciones presentes como las pasadas del sistema, junto a un modelo de evolución en el tiempo. Se usa el modelo para *predecir* el estado actual, usando estimaciones a priori de los estados como la condición inicial (lo cual incorpora información de datos pasados), de forma que se usan datos actuales para corregir la predicción anterior de la estimación del estado actual. Los algoritmos genéticos (GA) han sido ampliamente usados para el ajuste de parámetros tomando en cuenta una función de costo adecuada, por ejemplo, en [7], un método basado en algoritmos evolutivos fue desarrollado para encontrar los parámetros de embebimiento óptimos y aplicar la predicción de series de tiempo caóticas como función de costo para construir el modelo no lineal de la base de datos como se propone en [8].

En este estudio, se propone el uso del filtro de Kalman para predecir puntos en el atractor reconstruido, de forma que se construye una función de evaluación que cuantifica la capacidad de predicción de un atractor con el fin de valorar el ajuste de los parámetros de embebimiento. La comparación entre el atractor original y el predicho se realiza mediante medidas estadísticas. Así, se propone una metodología que facilitará la reconstrucción de atractores difeomórficos a los que se reconstruirían si se tuviera el modelo matemático de los sistemas fisiológicos subyacentes, a partir de series de tiempo unidimensionales.

II. MATERIALES Y MÉTODOS

A. Bases de datos

1) *Señales electrocardiográficas*: La base de datos MIT-BIH contiene 50 registros ECG de 2 canales, los cuales fueron digitalizados a 360 muestras por segundo, 11 bits de resolución y rango de 10 mV. La población muestral corresponde a 25 hombres entre 32 y 89 años y 25 mujeres con edades entre 23 y 89 años. En muchos registros, la señal superior (primer canal) es una modificación de la rama delantera II (MLII). La señal baja (segundo canal) es usualmente una modificación delantera de V1 (ocasionalmente V2 o V5) [9].

2) *Señales electroencefalográficas*: Los datos utilizados para las pruebas fueron recolectados por La Clínica para Epilepsia de la Universidad de Bonn [10]. La base de datos consta de cinco conjuntos (denotados A-E) compuestos por 100 segmentos de EEG de un canal. Los conjuntos A y B consisten de segmentos tomados de registros EEG superficiales (cuero cabelludo) en cinco personas saludables usando el esquema estándar de localización de electrodos 10–20. Los voluntarios se encontraban relajados en estado de vigilia con los ojos abiertos (A) y los ojos cerrados (B), respectivamente. Los conjuntos C, D y E se originan de registros EEG de diagnóstico prequirúrgico. Los segmentos en el conjunto D fueron registrados dentro de la zona epileptopatógena, y los del conjunto C de la formación hipocámpal del hemisferio opuesto del cerebro. Mientras que los conjuntos C y D contienen solamente actividad medida durante los intervalos interictales, el conjunto E contiene solamente actividad ictal. Todas las señales EEG fueron registradas con un sistema de adquisición de 128 canales, utilizando una referencia común promediada. Los datos fueron digitalizados a 173.61 Hz con una resolución de 12 bits.

B. Embebimiento y difeomorfismo

A partir de una serie de tiempo y_k , es posible reconstruir n puntos de un atractor $\mathbf{X} = \{\mathbf{x}_k\}_{k=1}^n$, con la expresión:

$$\mathbf{x}_k = [y_k \ y_{k+\tau} \ \dots \ y_{k+(m-1)\tau}] \quad (1)$$

donde m es la dimensión de embebimiento y τ el tiempo de retardo. La reconstrucción del atractor mediante (1) permite conocer las propiedades dinámicas del sistema que ha dado origen a la serie de tiempo y_k . Luego, bajo condiciones generales, el espacio de vectores \mathbf{x}_k generado por la dinámica contiene toda la información de la señal unidimensional y_k . El mapeo entre y_k y \mathbf{x}_k es suave e invertible. Esta propiedad indica su *difeomorfismo* y el mapeo su *embebimiento*. El teorema de embebimiento establece que con la serie de datos unidimensional de un sistema dinámico es posible reconstruir un espacio de estados difeomórfico al que se construiría si se tuvieran todas las variables dinámicas. Así, *difeomorfismo* significa que las propiedades cualitativas del atractor generado son iguales a las del original [11].

C. Filtro de Kalman

La evolución del atractor \mathbf{X} se define a través de un mapeo $\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k)$ que es desconocido. Con el fin de conocer el comportamiento futuro del atractor, es necesario conocer el mapeo $\mathbf{f}(\cdot)$ junto con los parámetros del atractor m y τ . Suponiendo que se tienen las cantidades m y τ , la función $\mathbf{f}(\cdot)$ puede ser reemplazada por un valor estimado $\hat{\mathbf{f}}(\mathbf{x}, \mathbf{w})$ dependiente del vector de parámetros \mathbf{w} .

El filtro extendido de Kalman (EFK) ha sido frecuentemente usado como una técnica para desarrollar estimaciones recursivas. Ya que este algoritmo sólo provee una *aproximación* de la estimación no lineal óptima, en [12], se presenta un filtro alternativo con un rendimiento superior al del EKF conocido como *filtro de Kalman unscented* (UKF). La estimación de los parámetros con el UKF consiste en realizar el aprendizaje de

un mapeo no lineal $\hat{\mathbf{f}}(\mathbf{x}, \hat{\mathbf{w}})$, donde \mathbf{w} corresponde al conjunto de parámetros desconocidos. El UKF puede ser utilizado para estimar los parámetros al llevar este problema dentro de la siguiente representación en espacio de estados [12]

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \boldsymbol{\omega}_k; \quad y_k = \mathbf{f}(\mathbf{x}_k, \mathbf{w}_k) + \nu_k \quad (2)$$

donde \mathbf{w}_k corresponde a un proceso estacionario con matriz de transición de estados identidad, alimentado por un ruido de proceso $\boldsymbol{\omega}_k \sim \mathcal{N}(0, \mathbf{R})$. La salida deseada y_k corresponde a una observación no lineal de \mathbf{w}_k considerando un ruido de medición ν_k .

D. Distancias estadísticas

1) *Distancia de Kolmogorov–Smirnov (DKS)*: Con esta distancia se busca la mayor desviación entre la función de distribución de la señal de prueba y_p y la señal estimada \hat{y} . Aunque también indica el grado de concordancia entre la distribución de probabilidad de la señal original y la distribución de la estimada. Esta medida está dada por:

$$D_{KS} = \max[\text{abs}(F_Y(y) - F_X(x))] \quad (3)$$

donde $F_X(x)$ y $F_Y(y)$ son las funciones de distribución de y_p y de \hat{y} respectivamente.

2) *Divergencia de Kullback–Leibler (DKL)*: Valora la diferencia entre dos distribuciones de probabilidad: desde una distribución de probabilidad *verdadera* P hasta una distribución de probabilidad *arbitraria* Q . En este caso P es la distribución de probabilidad de la señal y_p denotada como f_{y_p} y Q es la distribución de probabilidad de la señal \hat{y} denotada como $f_{\hat{y}}$ y la divergencia puede ser expresada como:

$$D_{KL} = \int P(x) \ln \frac{P(x)}{Q(x)} dx \quad (4)$$

3) *Distancia de Hellinger (DH)*: A diferencia de las demás, esta distancia es considerada matemáticamente como una métrica y está dada por:

$$D_H = \int [\sqrt{f_X(x)} - \sqrt{f_Y(y)}]^2 dx \quad (5)$$

E. Procedimiento

Para la realización de las pruebas se tomó un conjunto de entrenamiento compuesto por 50 registros ECG cada uno con una longitud de 4000 puntos y 50 registros EEG cada uno con una longitud de 4097 puntos, y se notan como $y(t)$. La función $\mathbf{f}(\cdot)$ está dada por un perceptrón multicapa recurrente (RMPL) con configuración $m - 8R - 7R - 1$ siendo m (la dimensión de embebimiento) el número de capas de entrada, $8R$ y $7R$ las capas ocultas y una capa de salida con funciones de activación sigmoideas. El entrenamiento se realiza en 10 épocas. El procedimiento propuesto para la estimación de los parámetros m y τ consiste en la estimación de la función $\hat{\mathbf{f}}(\mathbf{x}, \mathbf{w})$ para un m y τ dados y evaluación de la capacidad de predicción de esta función. Este procedimiento se realiza para diferentes valores de m y τ , hasta encontrar un valor para el cual se maximice la capacidad de predicción de $\hat{\mathbf{f}}(\cdot)$ sobre el atractor. El procedimiento se describe a continuación:

- 1) **Inicialización:** La serie de tiempo $\{y_k\}_{k=1}^N$ se separa en un conjunto de entrenamiento $y_e = \{y_k\}_{k=1}^{N_e}$ de 1200 puntos de cada registro y un conjunto de prueba $y_p = \{y_k\}_{k=N_e}^N$, con los puntos restantes.
- 2) **Selección de parámetros:** Se seleccionan los parámetros m y τ .
- 3) **Estimación:** Mediante el UKF se estiman los parámetros \hat{w} del mapeo $\hat{f}(x, w)$ sobre el atractor reconstruido usando (1) a partir del conjunto de entrenamiento y_e .
- 4) **Predicción:** A partir del mapeo estimado $\hat{f}(x, \hat{w})$ se predicen los valores futuros \hat{y} de y en el intervalo $[N_e, N]$.
- 5) **Evaluación de desempeño:** Se mide la similaridad entre la señal de prueba y_p y la estimada \hat{y} mediante las medidas de Kolmogorov–Smirnov, Kullback–Leiber y Hellinger [13].

III. RESULTADOS Y DISCUSIÓN

Inicialmente, se presentan los resultados de la metodología propuesta aplicada a una señal conocida (i.e., señal de Lorenz) para verificar la capacidad de predicción del atractor reconstruido. Las pruebas mostraron que una mala elección de los parámetros de embebimiento genera grandes diferencias entre la señal predicha \hat{y} y la señal de Lorenz y_p . Para la señal de Lorenz se encontró que la pareja (m, τ) adecuada es $(3, 2)$. En la Fig. 1 se muestra en rojo la señal de prueba y_p y en azul la señal predicha \hat{y} y se observa en la Fig. 1(a) que con un valor de τ erróneo la señal predicha \hat{y} es altamente afectada. Cuando el número de épocas es muy pequeño los resultados no son buenos como se aprecia en la Fig. 1(b), ya que experimentalmente se encontró que el transitorio que tienen las señales empieza a desaparecer después de 5 épocas. Se requirieron 10 épocas para la convergencia del algoritmo, mientras tanto se observaba la disminución del error cuadrático medio (MSE) entre y_p y \hat{y} (ver Fig. 2). En las Figs. 1, 3 y

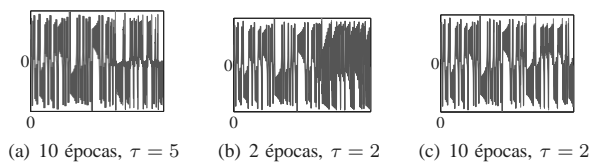


Fig. 1. Predicción de la señal de Lorenz con $m = 3$

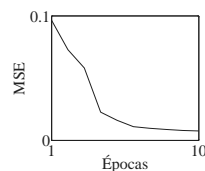


Fig. 2. Error cuadrático medio para la señal de Lorenz

4 se observa el efecto de una mala elección de los parámetros de embebimiento sobre la señal predicha \hat{y} . De las tablas I, II y III, se puede decir lo siguiente:

TABLA I
ERROR DE PREDICCIÓN PARA LAS PAREJAS $(3, \tau)$ CON LA SEÑAL DE LORENZ

τ	DKS	DKL	DH
2	0.03	0.15	0.08
5	0.04	0.24	0.17

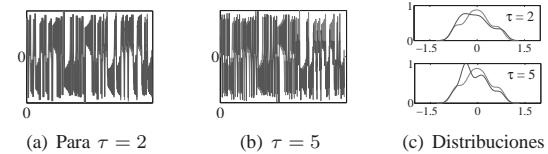


Fig. 3. Predicción de la señal de Lorenz con $m = 7$

TABLA II
ERROR DE PREDICCIÓN PARA LAS PAREJAS $(7, \tau)$ CON LA SEÑAL DE LORENZ

τ	DKS	DKL	DH
2	0.04	0.16	0.13
5	0.08	0.44	0.38

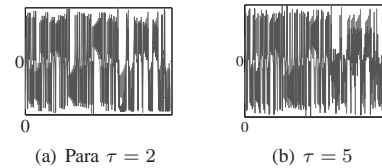


Fig. 4. Predicción de la señal de Lorenz con $m = 14$

TABLA III
ERROR DE PREDICCIÓN PARA LAS PAREJAS $(14, \tau)$ CON LA SEÑAL DE LORENZ

τ	DKS	DKL	DH
2	0.06	0.27	0.4
5	0.05	0.15	0.25

- Valores pequeños de la DH indica una buena elección de los parámetros de embebimiento.
- Debido a que la distribución de probabilidad de la señal y_p y \hat{y} no cambia significativamente al variar los parámetros de embebimiento (ver Fig. 3(c)), tampoco hay variación en DKS, luego esta medida no es relevante para evaluar la capacidad de predicción del atractor.

TABLA IV
ERROR DE PREDICCIÓN USANDO SEÑALES ECG Y EEG PARA $(3, 2)$

Señal	DKS	DKL	DH
ECG	0.22 ± 0.19	1.48 ± 1.17	13.09 ± 16.94
EEG	0.09 ± 0.04	0.19 ± 0.11	1.85 ± 1.43

En la tabla IV se muestran los valores de la media y la desviación estándar $\mu \pm \sigma$ de las 3 distancias consideradas en este estudio.

Después de realizar una comparación entre la información proveniente de las tres medidas estadísticas mediante la observación de los resultados presentados en las Figs. 5, 6, 7, 8, 9 y 10, se encontraron los siguientes aspectos a considerar:

- Valores mayores a la media en la DKS indican que la

señal de prueba y_p y la predicha \hat{y} son muy similares.

- En la DKL son los valores muy cercanos a la media los que indican similitud entre las dos señales.
- Para la DH son los valores por debajo de la media los que representan similitud entre las señales.

De los resultados observados en este estudio se puede decir que la DH ofrece información relevante para evaluar la capacidad de predicción del atractor. Mientras los valores sean menores a la media se percibe la similitud entre las señales y cuando los valores comienzan a tender a cero se obtienen buenas condiciones de embebimiento (i.e. buena elección de m y τ), como se observa en la Tabla I.

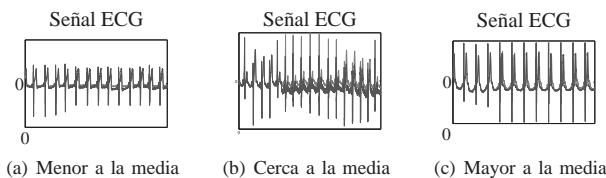


Fig. 5. Gráficas con diferentes valores de la DKS

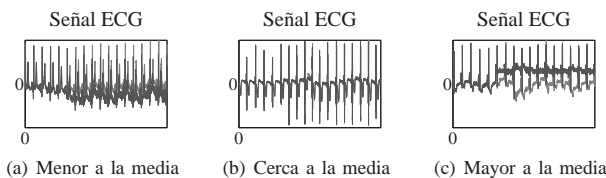


Fig. 6. Gráficas con diferentes valores de la DKL

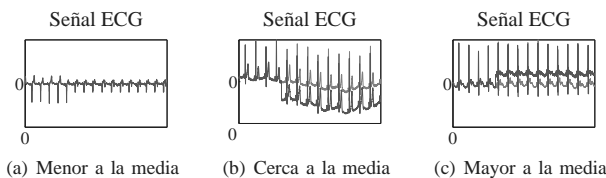


Fig. 7. Gráficas con diferentes valores de la DH

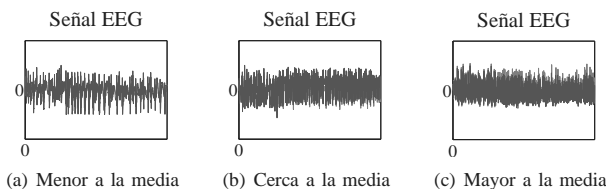


Fig. 8. Gráficas con diferentes valores de la DKS

IV. CONCLUSIONES

Se obtuvo un criterio para medir la calidad de predicción de un atractor basado en medidas estadísticas con el fin de evaluar el difeomorfismo de atractores obtenidos a partir de señales unidimensionales provenientes de sistemas, de los cuales no se conoce ningún modelo matemático consistente, como es el caso de la mayoría de los sistemas fisiológicos.

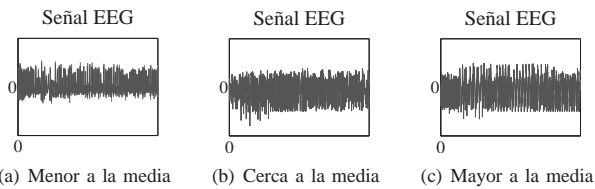


Fig. 9. Gráficas con diferentes valores de la DKL

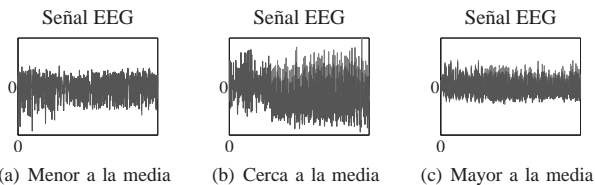


Fig. 10. Gráficas con diferentes valores de la DH

Con base en los resultados de este trabajo se puede afirmar que el buen ajuste de los parámetros de embebimiento aseguran que el atractor reconstruido captura la información intrínseca del sistema dinámico.

V. AGRADECIMIENTOS

Este trabajo se enmarca en el proyecto “Técnicas de computación de alto rendimiento en la interpretación automatizada de imágenes médicas y bioseñales” financiado por la DIMA y la Vicerrectoría de investigaciones de la Universidad Nacional de Colombia, con código 20201004224.

BIBLIOGRAFÍA

- [1] T. Sauer, J. Yorke, and M. Casdagli, “Embedology,” *Journal of Statistical Physics*, vol. 65, p. 579–616, 1991.
- [2] S. Jiang, F. Yang, P. Yi, B. Chen, M. Luo, and L. Wang, “Analysis of chaos attractors of MCG-recordings,” in *Proceedings of the 28th IEEE EMBS Annual International Conference*, 2006, pp. 6153–6156.
- [3] F. Takens, *Dynamical Systems and Turbulence, Lecture Notes in Mathematics*. Springer-Verlag, 1981, vol. 898, ch. Detecting strange attractors in turbulence, p. 366–381.
- [4] N. H. Packard, J. P. Crutchfield, J. D. Farmer, and R. S. Shaw, “Geometry from a time series,” *Physical Review Letters*, no. 45, pp. 712–716, 1980.
- [5] M. Behzad, H. Salarieh, and A. Alasty, “Chaos synchronization in noisy environment using nonlinear filtering and sliding mode control,” *Chaos, Solitons and Fractals*, vol. 36, p. 1295–1304, 2008.
- [6] B. R. Hunt, E. J. Kostelich, and I. Szunyogh, “Efficient data assimilation for spatiotemporal chaos: A local ensemble transform kalman filter,” *Physica D*, vol. 230, p. 112–126, 2007.
- [7] V. Babovic, M. Keijzer, and M. Stefansson, “Chaos theory, optimal embedding and evolutionary algorithms,” Danish Technical Research Council (STVF), Tech. Rep. 9800463, 2001.
- [8] S. A. Sannasiraj and V. Babovic, “Error forecasting in a wave prediction model using local linear model,” in *The Eighth Workshop on Ocean Models for the APEC Region*, Hong Kong, China, August 2002.
- [9] G. B. Moody and R. G. Mark, “The MIT-BIH arrhythmia database on CD-ROM and software for use with it,” in *Computers in Cardiology*, 1991, p. 185–188.
- [10] R. G. Andrzejak, K. Lehnertz, F. Mormann, C. Rieke, P. David, and C. E. Elger, “Indications of nonlinear deterministic and finite dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state,” *Physical Review E*, vol. 64, 2001.
- [11] P. Grassberger and I. Procaccia, “Measuring the strangeness of strange attractors,” *Physica D*, vol. 9, pp. 189–208, 1983.
- [12] S. Haykin, *Kalman Filtering and Neural Networks*. Wiley Interscience, 2001.
- [13] J. E. Hurtado and A. H. Barbat, *Stochastic Analysis of Multivariate Systems in Computational Mechanics and Engineering*. CIMNE, 1999, ch. Advanced topics in random processes, pp. 173–194.

Prototipo optoelectrónico basado en un correlador de transformada conjunta para verificación de huellas dactilares

Omar Javier Tijero Rojas, Yezid Torres Moreno, Miembro IEEE y Henry Argüello Fuentes*

Resumen. Este artículo reporta la construcción de un prototipo híbrido, óptico-digital, basado en la correlación de transformada conjunta para verificación de huellas dactilares. El dispositivo usa en la primera etapa una transformación de Fourier óptica con una pantalla de cristal líquido como soporte físico de entrada. El espectro de energía conjunta, JPS, adquirido por una cámara CCD es enviado a un procesador digital de señales, DSP, para procesamiento final. El desempeño del dispositivo y resultados de verificación de huellas dactilares son presentados.

Palabras claves: JTC, procesador híbrido, DSP, CCD, LCD, verificación, C++.

I. INTRODUCCIÓN

En el presente trabajo se presentan los resultados obtenidos en la evaluación de un prototipo optoelectrónico, construido para la verificación de huellas dactilares. Este dispositivo está compuesto por un correlador de transformada conjunta óptico y un procesador digital de señales donde se ejecutan los algoritmos para efectuar la comparación de las huellas.

Los umbrales para obtener criterios de comparación son hallados según la probabilidad de encontrar en una población un impostor que quiera suplantar a un individuo con éxito (error tipo I llamado falsa aceptación), y un individuo que quiera verificarse y sea rechazado (error tipo II llamado falso rechazo) [5].

De manera semejante, se describe cada etapa del prototipo para el entendimiento de su funcionamiento, se hacen las pruebas indispensables a cada uno de los elementos para que el funcionamiento de las etapas opere en su región óptima lineal y finalmente, se reportan las diferentes medidas espaciales, temporales y los errores de verificación.

II. MARCO TEORICO Y GENERALIDADES

Los cálculos para el diseño y los avances teóricos básicos que se utilizaron para la construcción del prototipo corresponden a medidas de desempeño que evalúan un correlador tales como PSR, SNR y PCE (obsérvese un resumen en las ecuaciones 1-5 apoyadas en la figura 1) [8]. De la ecuación 1 a la 4, se describe paso a paso la operación de correlación (1 a 3 ópticamente y 4 en un procesador digital) y la ecuación 5 representa una de las restricciones más significativas en el diseño del prototipo.

$$u_1(x_1, y_1) = f(x_1, y_1 - Y/2) + g(x_1, y_1 + Y/2) \quad (1)$$

$$u_2(x_2, y_2) = \frac{1}{\lambda f} F\left(\frac{x_2}{\lambda f}, \frac{y_2}{\lambda f}\right) \exp\left(-j\pi y_2 \frac{Y}{\lambda f}\right) + \frac{1}{\lambda f} G\left(\frac{x_2}{\lambda f}, \frac{y_2}{\lambda f}\right) \exp\left(j\pi y_2 \frac{Y}{\lambda f}\right) \quad (2) [3]$$

$$I(x_2, y_2) = |u_2(x_2, y_2)|^2 = \frac{1}{(\lambda f)^2} \left[\left| F\left(\frac{x_2}{\lambda f}, \frac{y_2}{\lambda f}\right) \right|^2 + \left| G\left(\frac{x_2}{\lambda f}, \frac{y_2}{\lambda f}\right) \right|^2 + F\left(\frac{x_2}{\lambda f}, \frac{y_2}{\lambda f}\right) G^*\left(\frac{x_2}{\lambda f}, \frac{y_2}{\lambda f}\right) e^{-j2\pi y_2 \frac{Y}{\lambda f}} + F^*\left(\frac{x_2}{\lambda f}, \frac{y_2}{\lambda f}\right) G\left(\frac{x_2}{\lambda f}, \frac{y_2}{\lambda f}\right) e^{j2\pi y_2 \frac{Y}{\lambda f}} \right] \quad (3)$$

$$u_3(x_3, y_3) = \frac{1}{\lambda f} \left[\begin{aligned} & f(x_3, y_3) * f^*(-x_3, -y_3) + g(x_3, y_3) * g^*(-x_3, -y_3) \\ & + f(x_3, y_3) * g^*(-x_3, -y_3) * \delta(x_3, y_3 - Y) \\ & + f^*(-x_3, -y_3) * g(x_3, y_3) * \delta(x_3, y_3 + Y) \end{aligned} \right] \quad (4)$$

$$Y > \max\{Ry_1, Ey_1\} + \frac{Ry_1 + Ey_1}{2} \quad (5) [3]$$

Los pasos que se siguieron para la construcción del prototipo están ceñidos a la figura 1 y son los listados a continuación.

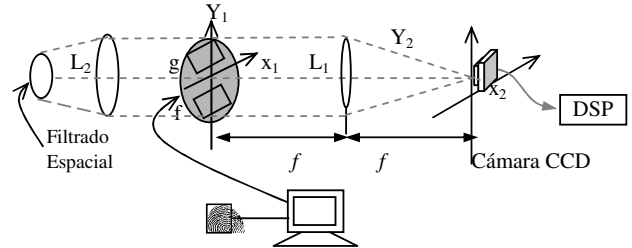


Figura 1. Diseño general del prototipo. Referencia [8].

A. ETAPA DE PROCESAMIENTO ÓPTICO

Para la construcción del prototipo óptico se tiene en cuenta el diseño (ecuaciones 1-5) y la caracterización de los dispositivos presentados en [8], además de algunos valores por defecto dados por los fabricantes de los instrumentos utilizados.

1) Filtro Espacial. Es uno de los componentes más importantes del prototipo ya que es la región donde se filtran componentes espaciales del haz de luz de alta frecuencia que incide en el procesador, y está compuesto por un objetivo de microscopio de 40X y un pinhole de 10µm; apto para un láser de una longitud de onda de 532 nm. En esta etapa se siguen las instrucciones dadas por el fabricante como lo muestra [6] y la figura 4.

2) Colimación. Es la etapa del procesado en la que se obtiene un haz de luz plano y es necesaria para que el tamaño de las imágenes proyectadas no sufran, en lo posible,

* Grupo de Óptica y Tratamiento de Señales. Universidad Industrial de Santander. AA. 678. Bucaramanga, Colombia. otijero@gmail.com, ytorres@uis.edu.co, henarfu@uis.edu.co

escalamiento en su procesado. Este se forma con una lente que por lo general es diseñada para tal caso. En la figura 2 se muestra el ejemplo de una onda colimada donde la distancia focal f en el caso del prototipo es de 38 cm. y el diámetro D de la lente es de 8 cm.

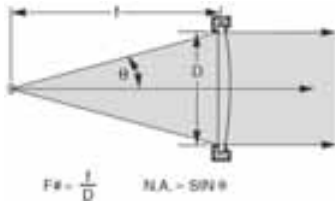


Figura 2. Colimación de la onda generada por un rayo láser previamente filtrada espacialmente. Referencia: [7].

3) Plano de entrada. Es el lugar en el que se despliegan las imágenes a procesar y está formado principalmente por la pantalla de cristal líquido (traslúcida) y dos polarizadores ópticos que permiten el funcionamiento de la misma en modo de amplitud. En esta etapa se hicieron pruebas, para trabajar en la región óptima lineal que, finalmente fue de 0° para el polarizador y 36° para el analizador. Se denomina polarizador al dispositivo polarizador colocado antes de la LCD (izquierda a derecha) y analizador al dispositivo de polarización gemelo ubicado después.

4) Transformación de Fourier. En esta etapa se utiliza una lente de Fourier a una distancia de 37.15 cm. según los cálculos realizados en [8], y se efectúa la debida transformación la cual se sensó en una cámara CCD. Aquí finaliza el procesamiento óptico pues la imagen es capturada y procesada digitalmente por un DSP embebido en una tarjeta de desarrollo para tratamiento de imágenes.

B. ETAPA DE PROCESAMIENTO DIGITAL

Para el procesamiento de las imágenes adquiridas en la cámara CCD se utiliza un DSP y, para generar las imágenes del plano de entrada se simula un servidor que registra al usuario de ingreso y compone una nueva imagen con un conjunto de posibles coincidencias (los registros almacenados en la base de datos del usuario registrado).

1) Procesado en DSP. Para el desarrollo de los algoritmos de decisión, se utiliza, en la captura de las imágenes procesadas ópticamente, un DSP que tiene la función de tomar la densidad espectral conjunta de energía y realizar una transformada de Fourier, la cual arrojó en los casos de la distribución genuina picos de correlación altos y en la distribución impostora, picos poco notables.

III. PRUEBAS Y RESULTADOS

A. IMÁGENES DEL PLANO DE ENTRADA.

Para el diseño de la simulación de un servicio que colocase imágenes de entrada se utilizó una versión de prueba de Adobe Flash [1]. Para escoger la ubicación espacial se tuvieron en cuenta los conceptos de traslape que se encuentran dados en la ecuación 5. Allí se utilizó un espacio dedicado para la búsqueda en la base de datos (superior derecho) y otro espacio para la imagen adquirida del sensor de huellas dactilares

(inferior izquierdo), el cual podría ser habilitado dentro del mismo software para la adquisición de nuevas imágenes para el procesado (como en un prototipo automático), no para almacenar en base de datos.

B. IMÁGENES EN EL SENSOR CUADRÁTICO CCD.

Para comprobar que el sensor funcionase bien se hicieron pruebas a partir de 50 imágenes que se adquirieron para cada prueba. Las medidas utilizadas fueron la energía de la imagen, los valores de los píxeles (la imagen como un todo) y luego por la toma de píxeles aleatorios (ambas en el transcurso del tiempo); sin embargo como el sensor es a color y su formato es NTSC, la imagen se forma en 3 planos. Los cálculos fueron hechos a partir de la razón entre la media y la varianza de las imágenes o de los píxeles. A esta razón se le ha llamado SNR en [4] y es derivado del parámetro de la correlación con el que se ha experimentado.

Las pruebas se realizaron primero sin ninguna iluminación y luego con iluminación en lo posible uniforme (sin llegar a la saturación). Se tomaron 12 píxeles aleatorios y el cambio de su relación señal a ruido (SNR). En la figura 3 se muestra una figura comparativa de los promedios en los 3 niveles (RGB) de la energía en los píxeles (mostrados en dB) de las 50 imágenes, y a continuación se muestran los promedios de las imágenes totales.

Sin iluminación:	R	G	B
Energía Imagen -> SNR:	73.7430	73.0011	73.2853
Valor Imagen -> SNR:	73.5729	74.5453	68.2199
Iluminación uniforme:	R	G	B
Energía Imagen -> SNR:	97.7332	84.3744	83.1822
Valor Imagen -> SNR:	103.4223	94.7736	91.7900

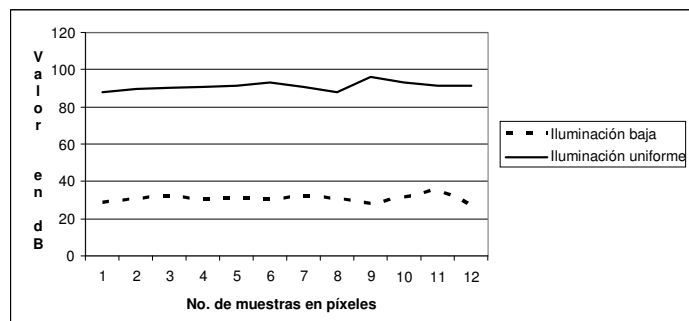


Figura 3. Comparación de la relación señal a ruido en la CCD por píxel.

Obtenidos los resultados, se pudo inferir que la cámara responde muy bien en la zona de iluminación cuasi uniforme, mientras que con poca luz la respuesta no es la más deseada.

C. DIMENSIONES FINALES.

El tamaño final del prototipo híbrido se describe en la figura 4.