

RISCE Revista Internacional de Sistemas Computacionales y Electrónicos

JULIO 2010
Número 4, Volumen 2, Año 2



RISCE Revista Internacional de Sistemas Computacionales y Electrónicos; es una publicación bimestral del Instituto Politécnico Nacional, Av. Luis Enrique Erro S/N, unidad “Profesional Adolfo López Mateos”, Del. Gustavo A. Madero, C.P. 07738, México D.F. a través de la Escuela Superior de Computo; Av. Juan de Dios Bátiz S/N esquina Miguel Othón de Mendizábal. “Unidad Profesional Adolfo López Mateos”. Col. Lindavista C.P. 07738, México, D. F. tel. 57296000 ext. 52000. Certificado de reserva de Derechos al uso Exclusivo del título No. 04-2008-062613190500-203, ISSN en trámite. Los artículos son responsabilidad exclusiva del autor y no reflejan necesariamente el criterio de la institución, a menos que se especifique lo contrario. Se autoriza la reproducción total o parcial, siempre y cuando se cite explícitamente la fuente.

La revista se especializa en el área de los sistemas computacionales y electrónicos; tanto en el desarrollo, como en la investigación en:

Ciencias de la Computación
Cómputo educativo
Cómputo Móvil
Comunicaciones
Disciplinas Emergentes
Electrónica
Física Electrónica
Ingeniería de Cómputo
Ingeniería de Software
Innovación Tecnológica
Inteligencia artificial
Matemática computacional
Procesamiento de señales
Robótica y cibernética
Sistemas de Información
Tecnologías de la Información

Distribución

La revista cuenta con 300 ejemplares que se distribuyen en:

Europa, Asia y América Hispana; mediante CD ROM y correo electrónico

Directorio



INSTITUTO POLITÉCNICO NACIONAL

DRA. YOLOXÓCHITL BUSTAMANTE DÍEZ
DIRECTORA GENERAL

ING. JUAN MANUEL CANTÚ ALVAREZ
SECRETARIO GENERAL

DR. EFREN PARADA ARIAS
SECRETARIO ACADEMICO

DR. JAIME ALVAREZ GALLEGOS
SECRETARIO DE INVESTIGACIÓN Y POSGRADO

ING. ERNESTO MERCADO ESCUTIA
SECRETARIO DE SERVICIOS EDUCATIVOS

ING. OSCAR JORGE SÚCHIL VILLEGAS
SECRETARIO DE EXTENSIÓN E INTEGRACION SOCIAL

M. EN C. FERNANDO ARELLANO CALDERON
SECRETARIO DE GESTION ESTRATEGICA

C.P. ROBERTO ALVAREZ ARGUELLES
SECRETARIO DE ADMINISTRACION

LIC. JUDITH CLAUDIA RODRIGUEZ ZUÑIGA
DEFENSORA DE DERECHOS POLITECNICOS



ESCUELA SUPERIOR DE CÓMPUTO

ING. APOLINAR FRANCISCO CRUZ LÁZARO
DIRECTOR

DR. FLAVIO ARTURO SÁNCHEZ GARFIAS
SUBDIRECTOR ACADÉMICO

DR. JESÚS YALJÁ MONTIEL PÉREZ
JEFE DE LA SECCIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

LIC. ARACELI LOYOLA ESPINOSA
SUBDIRECTORA DE SERVICIOS EDUCATIVOS E INTEGRACIÓN SOCIAL

M. EN C. JUAN VERA ROMERO
SUBDIRECTOR ADMINISTRATIVO

DR. EDUARDO BUSTOS FARÍAS
EDITOR DE RISCE

Miembros del comité Revisor

(Todo el comité técnico está formado por doctores en ciencias o su equivalente)

Francisca Losavio de Ordaz (Venezuela) (Universidad Central de Venezuela)

Alfredo Matteo (Venezuela) (Universidad Central de Venezuela)

Emmanuel F. Moya Anica (México)

Edgardo Manuel Felipe Riverón (Cuba) (México) (CIC)

Luis Enrique Palafox Maestre (México)

Eduardo F. Caicedo Bravo (Colombia)

Hilda Ángela Larrondo (Argentina)

Guillermo Leopoldo Kemper Vásquez (Perú)

Elizabeth León Guzmán (Colombia)

María Cecilia Rivera (Chile)

Satu Elisa Schaeffer (Finlandia) (UANL)

Rafael Canetti (Uruguay)

Javier Echaiz (Argentina)

Pablo Belzarena (Uruguay)

Carlos Beltrán González (Italia) (Università di Genova)

Elena Fabiola Ruiz Ledesma (México)

Jonatan Gómez (Colombia)

Armando De Giusti (Argentina)

Juan José Torres Manríquez (México)

Jesús Yaljá Montiel Pérez (México)

Luis Alfonso Villa Vargas (México)

Marco Antonio Ramírez Salinas (México)

Félix Moreno González (España) (UPM)

Salvador Godoy Calderón (México) (CIC)

INDICE

Acumulación de conocimiento que proviene de la Web.....	6
Alma Delia Cuevas Rasgado , Adolfo Guzmán Arenas	
Sistema de Cómputo Móvil para el control de la Diabetes Mellitus	11
Juan José Torres Manríquez., Ignacio Ríos de la Torre, José Luis. Calderón Osorno	
Apoyo en un Software Educativo para el desarrollo de competencias matemáticas en la Unidad Didáctica de Probabilidad y Estadística en el nivel Superior	16
Elena Fabiola Ruiz Ledesma	
Closed Form Equations for Low Order Wavelet FIR Filters	23
Alfonso Fernandez-Vazquez	
Pruebas de un algoritmo de compresión fractal basado en quadtree para imagen	33
Jesús Yaljá Montiel Pérez, Rodolfo Romero Herrera, José Luis López Bonilla	
Instrucciones para los autores	41

Acumulación de conocimiento que proviene de la Web

Alma Delia Cuevas Rasgado¹ , Adolfo Guzmán Arenas²

¹ Escuela Superior en Cómputo Av. Juan de Dios Batiz casi esq. Miguel Othon de Mendizabal Unidad Profesional Adolfo Lopez Mateos, Edificio ESCOM. Col. Nueva Industrial Vallejo, Delegación Gustavo A. Madero, C. P. 07738, Mexico D. F. f.almadeliacuevas@gmail.com

² Centro de Investigación en Computación , misma dirección. a.guzman@acm.org

Resumen.

Cuando una persona agrega nuevos conocimientos a los que ya posee, toma en cuenta información novedosa, detalles adicionales, mayor precisión, sinónimos, homónimos, redundancias, contradicciones aparentes e inconsistencias entre lo que ya sabe y el nuevo conocimiento que absorbe. Así adquiere incrementalmente información manteniendo siempre un conocimiento consistente. Queremos hacer lo mismo con computadoras. Las ontologías de la computadora son equivalentes a esa región cerebral de una persona que concentra y administra la información. Exponemos un algoritmo para fusionar dos ontologías (fusionar el “conocimiento” de dos máquinas). Anteriormente ya había algoritmos de este tipo pero carecían de características importantes, siendo meramente editores de ontologías que requerían de una persona para resolver los detalles. El algoritmo que se presenta se llama OM (Ontology Merging), y une o fusiona ontologías (provenientes de documentos en la Web) en forma automática (sin intervención humana), considerando las inconsistencias, contradicciones y redundancias entre ambas ontologías, de forma que el resultado sea lo más cercano a la realidad. OM obtiene buenos resultados, al compararlos contra uniones efectuadas manualmente. El uso repetido de OM permite adquirir gran información sobre un mismo tema.

Abstract.

When a person adds new knowledge they already have, consider new information, additional details, more precisely, synonyms, homonyms, redundancies, inconsistencies and apparent contradictions between what they already know and absorbing new knowledge. This information becomes incrementally maintaining a consistent knowledge. We do the same with computers. The computer ontologies are equivalent to that of an individual brain region that concentrates and manages information. We present an algorithm to merge two ontologies (to merge the "knowledge" of two machines). Previously he had algorithms of this type but lacked important features, being merely ontology editors that required a person to work out the details. The algorithm presented is called OM (Ontology Merging), and joining or merging ontologies (from Web documents) automatically (without human intervention), given the inconsistencies, contradictions and overlaps between both ontologies, so that the result is as close to reality. OM achieves good results, when compared against unions carried out manually. Repeated use of OM can acquire great information about a topic.

1 Introducción

Una persona acumula información a lo largo de su vida al ir añadiendo conocimiento (conceptos, relaciones, valores típicos.) nuevo al que ya posee en su mente (en su “ontología” o estructura del conocimiento), identificando redundancias, información nueva, pequeñas contradicciones, contradicciones severas, sinónimos y antónimos, entre otros casos. Hasta ahora, la computadora podía hacer el mismo proceso (unir conocimientos provenientes de dos fuentes u ontologías distintas) usando un editor que le facilita la tarea a una persona, quien finalmente decide. El problema a resolver es cómo automatizar esa unión.

Se presenta un algoritmo (OM, Ontology Merging) y su implementación, para fusionar dos ontologías en forma automática, obteniendo una tercera, considerando las inconsistencias, sinonimias, grado de precisión, contradicciones y redundancias entre ambas, de tal manera que el resultado sea lo más cercano a la realidad.

La ontología resultante puede convertirse en una ontología del conocimiento actual si se fusionan varias ontologías de propósito general y específica.

Los trabajos actuales se muestran en la sección 1.2; un ejemplo está en la sección 4. Más ejemplos en [2]

De esta manera los usuarios de OM* podrán obtener respuestas concretas de OM* y no solo pedirle (como se le pide a Wikipedia o a Google) que proporcione documentos sobre ciertos temas y el usuario haga la extracción o deducción.

La propuesta se parece al Proyecto CYC [11], que buscó, durante una década, fabricar (manualmente) una ontología del conocimiento común.

1.2 Trabajos relacionados

A diferencia de la creación manual de ontologías, como en: [11], el equipo de trabajo de este artículo está atacando el problema por otro camino: la creación de la ontología del conocimiento de manera automática, obteniendo pedacitos de conocimiento (ontologías chicas) y uniéndolos con cuidado (filtrando las inconsistencias, uniendo sinónimos...).

Los métodos actuales de fusión de ontologías, como. PROMPT expuesto en [4], o Chimaera [10], requieren que el usuario resuelva los problemas presentados durante la fusión. Otros como IF Map[8], FCA-Merge [10] usan el Análisis Formal de Conceptos para la representación de sus ontologías, forzando a éstas ser consistentes (pero la mayoría de las ontologías en la Web suelen presentar inconsistencias). Destaca el algoritmo de fusión HCONE-merge [9], pues utiliza la base de datos semántica WordNet [4] como información intermediaria para la fusión, requiriendo menos del apoyo del usuario, lo cual significa un importante avance hacia la unión automática de ontologías.

Nuestro propósito es usar OM para que la misma máquina vaya juntando ontologías pequeñas que encuentre, para ir formando (lo más coherentes entre sí) otras más grandes, acumulando así inmenso conocimiento. ¡Interesante y no menos desafiante!.

2 Concepto de Fusión de Ontologías

Una ontología es una tupla $O = (\mathcal{C}, R)$ donde:

$\mathcal{C}$ es un conjunto de nodos (que denotan conceptos) de los cuales algunos de ellos son relaciones. Donde: R es un conjunto de restricciones, de la forma $(r; c_1; c_2; \dots; c_k)$ entre la relación r y los conceptos c_1 hasta c_k (se usa c minúscula para referirse a cada concepto del conjunto $\mathcal{C}$ y punto y coma para separar los argumentos). Las relaciones no están limitadas a dos argumentos. Nótese que una ontología es un hipergrafo con nodos del conjunto $\mathcal{C}$ e hiper-relaciones R .

Considerando dos ontologías A y B , se trata de unir las en una tercera ontología C (formando una nueva ontología), de manera general se muestra como:

$$C = A \cup \{cC \mid cC = \text{ext}(rA, rB) \ \forall \ cA \in A, cB \in B, cc \in C, rA, rB \in R\}$$

La ontología resultante C es la ontología original A añadida de ciertos conceptos y relaciones de B que la función ext extrae.

Donde:

cA es un concepto de la ontología A , rA son las relaciones de cA que existen en A , rB son las relaciones de cB que existen en B y cB es el concepto más similar c_{ms} en B a cA ;

$\cup \square$ indica una unión de ontologías. Es una unión “cuidadosa” y no la unión de conjuntos.

$\text{ext}(rA, rB)$ es el algoritmo que complementa las relaciones rA que ya están en C con aquellas de cB (que están en B) que no contradicen el conocimiento de A .

Al aplicar ext a cada concepto cA de A , el algoritmo OM extrae de B el conocimiento “adicional” que no estaba presente en A y lo agrega al resultado C . Esta extracción debe hacerse con cuidado, para no introducir inconsistencias, contradicciones o información redundante en C . El algoritmo ext es extenso, una parte se explica en la sección 4.

3 Elementos que integran a OM

OM se apoya de algunas bases de conocimiento y recursos que le ayudan a detectar contradicciones, encontrar sinónimos, etc. Estas son:

- 1 artículos y conectores tales como (en, el, para, este, y, o, etc.) que son ignorados en el nombre de las relaciones.
- 2 palabras que cambian o niegan la presencia de conceptos en los nombres de las relaciones, tales como: excepto, sin. Por ejemplo: Amapola sin Peciolo. Significa que el concepto Peciolo no forma parte del concepto Amapola.
- 3 Una jerarquía de conceptos. Esta jerarquía es como un árbol de conceptos donde cada nodo es un concepto en otros casos es un conjunto de conceptos de un mismo tipo y en otros es una partición de conceptos (los conceptos que forman la partición son mutuamente exclusivos y juntos forman la partición). La jerarquía representa una taxonomía de términos relacionados entre sí y se usan para medir la confusión [6] y después pueden ser usados para la detección de sinónimos.

4 Método OM para Fusionar Ontologías

De manera general se presenta el algoritmo de fusión OM. Sus entradas son A y B:

1. Copia la ontología A hacia C.
2. Partiendo del concepto raíz c_{Raiz} en C.
3. Busca en B su concepto más similar c_B (usando COM expuesto en [1]), el concepto más similar también es conocido como *cms*.
4. Si hay un *cms* en B se adicionan nuevas relaciones, nuevos conceptos, se verifican sinónimos, detectan y resuelven algunas inconsistencias (usando la teoría de la confusión [7]), se verifica e impide la copia de las relaciones redundantes.
5. Si no hay un *cms* accede al siguiente nodo c_C (hijo de la raíz) y regresa a 3

Si en el paso 4 no se resuelven las inconsistencias se conserva la relación inconsistente en C (la ontología resultante). Más detalles sobre este algoritmo se explican en [2].

OM se sustenta en dos trabajos importantes:

- El comparador COM [7], que toma un concepto c_A en una ontología A y halla el concepto más similar c_B en una ontología B.
- La teoría de la confusión [6], que obtiene el grado de confusión de usar un concepto r en lugar de otro concepto s y la confusión de usar s en lugar de r .

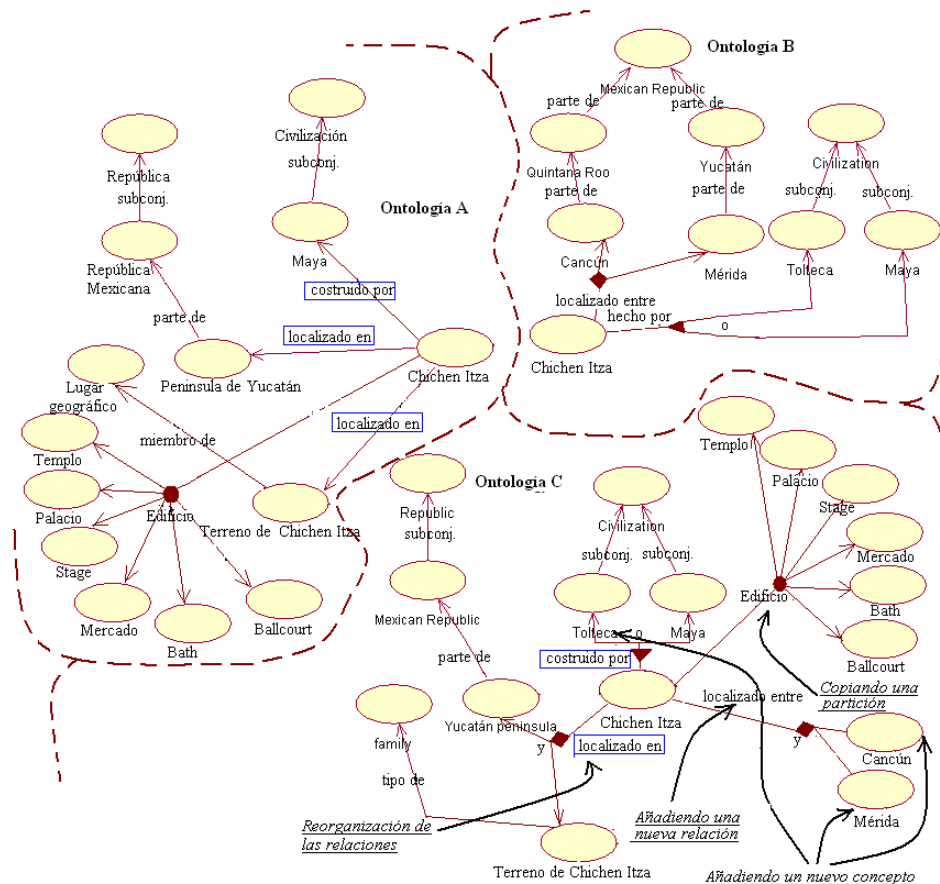


Figura 1. Muestra dos ontologías diferentes A y B que describe Chichen Itza y su fusión en C (solo se presenta las partes importantes). Las líneas discontinuas son las fronteras entre cada una de las parte de las tres ontologías. Las frases subrayadas son las indicaciones del proceso de fusión por ejemplo. De la frase: Añadiendo un nuevo concepto en la ontología C salen tres flechas que apuntan a los conceptos: Tolteca, Mérida y Cancún, significa que estos tres conceptos se han añadido a la ontología C resultante.

En la figura 1 se presenta el proceso de fusión en la que aparecen los siguientes casos. Existen más ejemplos en [2]:

1. Copia de particiones nuevas. Edificio es una partición en la ontología A (representado con un círculo pequeño) de Chichen Itza, lo cual es añadido a la ontología C.
2. Copiando nuevos conceptos. Los conceptos: *Tolteca*, *Mérida* y *Cancún* no estaban en A, aparecen en B. por lo tanto, son copiados por OM a C.
3. Reorganizacion de relaciones. La relación *localizado en* aparece dos veces en A pero con diferentes valores, por lo tanto se fusiona en C en una sola relación.

5 Conclusiones

Con la llegada de OM, es posible ahora fusionar dos ontologías de manera automática, sin intervención humana. Las pruebas efectuadas muestran calidad en los resultados obtenidos por OM. OM detecta inconsistencias y resuelve algunas, detecta sinónimos, homónimos, información redundante y con diferente grado de detalle o precisión.

Hacen falta un convertidor de documentos en lenguaje natural a ontologías, y un razonador que responda (usando la ontología resultado de OM) preguntas complejas.

Referencias

- [1] Cuevas, A., and Guzmán, A. "Improving the Search for the Most Similar Concept in other Ontology". In proc. XVIII Congreso Nacional y IV Congreso Internacional de Informática y Computación. Torreón Coah. México. Octubre 2005.
- [2] Cuevas, A. Merging of ontologies using semantic properties. Ph. D. thesis. CIC-IPN. 2006. In Spanish. <http://148.204.20.100:8080/bibliodigital/ShowObject.jsp?idobject=34274&idrepositorio=2&type=recipiente>
- [3] Alma-Delia Cuevas, Adolfo Guzman. (2008) A language and algorithm for automatic merging of ontologies. Chapter of the book Handbook of Ontologies for Business Interaction, Peter Rittgen, ed. IGI Global (formerly Idea Group Inc.), USA. 381-404
- [4] Fellbaum, C. "WordNet An Electronic Lexical Database". Library of Congress Cataloging in Publication Data. 1999.
- [5] Fridman, N., and Musen, M. "PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment". In Proc. Seventeenth National Conference on Artificial Intelligence. pp 450-455, Austin, TX, USA, 2000.
- [6] Guzmán, A., and Levachkine, S. "Hierarchies Measuring Qualitative Variables". Lecture Notes in Computer Science LNCS 2945 [Computational Linguistics and Intelligent Text Processing], Springer-Verlag. 262-274. ISSN 0372-9743. 2004.
- [7] Guzmán, A., and Olivares, J. "Finding the Most Similar Concepts in two Different Ontologies". Lecture Notes in Artificial Intelligence LNAI 2972, Springer-Verlag. 129-138. ISSN 0302-9743. 2004.
- [8] Kalfoglou, Y., and Schorlemmer, M. "Information-Flow-based Ontology Mapping". Proceedings of the 1st International Conference on Ontologies, Databases and Application of Semantics (ODBASE'02), Irvine, CA, USA. 2002.
- [9] Kotis, K., and Vouros, G., Stergiou, K. "Towards Automatic of Domain Ontologies: The HCONE-merge approach". Elsevier's Journal of Web Semantic (JWS), vol. 4:1, pp 60-79. Available on line ant (ScienceDirect): <http://authors.elsevier.com/sd/article/S1570826805000259> 2006.
- [10] McGuinness, D., Fikes, R., Rice, J., and Wilder, S. "The Chimaera Ontology Environment Knowledge". In Proceedings of the Eighth International Conference on Conceptual Structures Logical, Linguistic, and Computational Issues (ICCS 2000). Darmstadt, Germany. 2000.
- [11] Reed, S. L., and Lenat, D. "Mapping Ontologies into Cyc". In proceeding of AAAI Workshop on Ontologies and the Semantic Web, Edmonton, Canada. 2002.
- [12] Stumme, G., Maedche, A. "Ontology Merging for Federated ontologies on the semantic web". In: E. Franconi, K. Barker, D. Calvanese (Eds.): Proc. Intl. Workshop on Foundations of Models for Information Integration (FMII'01), Viterbo, Italy, 2001. INAI, Springer 2002 (in press).

Sistema de Cómputo Móvil para el control de la Diabetes Mellitus.

Juan J. Torres Manríquez., Ignacio Ríos de la Torre, José Luis. Calderón Osorno.

Escuela Superior de Cómputo del IPN
Av. Juan de Dios Bátiz S/N Miguel Othón de Mendizábal Col. Lindavista Del. G. A. Madero.
México DF. CP 07738
jtordesm@ipn.mx, irios@ipn.mx, jcalderono@ipn.mx

Resumen

En este trabajo diseñamos un sistema computacional para el pre-diagnóstico y control de la Diabetes Mellitus, por medio de la inteligencia artificial a través de un dispositivo móvil (PDA). Este sistema registra datos patológicos, clínicos y antecedentes familiares para generar como salida un pre-diagnóstico, para que el usuario esté informado si tiene la enfermedad, si no la tiene, si es propenso o está en riesgo de tenerla en un futuro. El sistema genera esta información por medio de tablas, gráficas y estadísticas para un correcto tratamiento y control de tal enfermedad con los avances tecnológicos actuales y además proporciona información acerca de dietas y recomendaciones.

Abstract.

In this paper we design a computer system for pre-diagnosis and control of Diabetes Mellitus, by means of artificial intelligence through a mobile device (PDA). This data system records pathological, clinical and family history to generate as output a pre-diagnosis, so that the user is informed if you have the disease, if it does not, if you are prone or at risk of having it in the future. The system generates this information through charts, graphs and statistics for treatment and control of the disease with current technological advances and also provides information about diet and recommendations.

Introducción.

Hoy en día la diabetes mellitus es una enfermedad de interés nacional ya que está afectando cada vez más a la población mexicana y es ya una de las principales causas de muerte en nuestro país, es por ello la importancia de tener conocimiento de esta enfermedad y más aun saber del tratamiento adecuado para tener controlada esta enfermedad. La población más afectada es la adulta mayor a los veinte años de edad.

La diabetes es una enfermedad ocasionada por la incapacidad del cuerpo de generar insulina o el cuerpo la utiliza incorrectamente, también ya se tiene clasificada la diabetes en tipos. En México el promedio de vida de un diabético mal controlado es de 57 años, es decir casi 20 años del promedio de la población general y también se estima que cada 2 horas muere una persona por complicaciones de la diabetes, pero es importante señalar que esta enfermedad es controlable al 100% y por ello es necesario conocer cuáles son los factores de riesgo que desarrolla la diabetes para llevar un tratamiento adecuado y tener un nivel de vida normal. La población que presenta esta tipo de enfermedad cambia drásticamente su nivel de vida y la de las personas que la rodean, ya que tiene que estar al pendiente de los cuidados necesarios para llevar un control de la enfermedad y tener un mejor nivel de vida.

Objetivo General.

*Diseñar e implementar un sistema de información el cual sirva para el pre diagnóstico y control de la diabetes Mellitus, por medio de la inteligencia artificial y dispositivo móvil (PDA).

Objetivos Específicos.

*Diseñar un sistema experto que por medio de la introducción de datos patológicos, clínicos y antecedentes familiares genere como resultado un pre-diagnostico, es decir que el usuario conozca si tiene la enfermedad si no la tiene, si es propenso o está en riesgo de tenerla en un futuro

*Diseñar un sistema de control que a partir de los datos clínicos introducidos por el paciente en este caso las medidas de las glucemias que arroja el glucómetro genere tablas, graficas y estadísticas para un mejor tratamiento y control de la enfermedad.

*Diseñar un sistema que tenga informado al paciente acerca de la Diabetes Mellitus, con los avances científicos y tecnológicos para el control de la misma, información acerca de dietas, recomendaciones, artículos, etc.

Métodos y Materiales.

Este sistema de información está montado en un dispositivo móvil para que el usuario en este caso el paciente siempre tenga acceso a éste. El sistema contiene información actualizada acerca de la diabetes mellitus, así como cuales son los síntomas, consecuencias y como afecta al cuerpo en general. Este sistema es un sistema experto que genera un pre- diagnostico utilizando los datos que el paciente introduzca. Además el sistema lleva un control de los niveles de azúcar que el usuario registra y muestra esos datos en gráficas cuando estos sean requeridos por el paciente, también el sistema genera y muestra un tratamiento de acuerdo a los datos que le sean solicitados al paciente, el tratamiento contiene una serie de consejos, así como también una dieta de alimentos que podría consumir. Un punto importante es que el sistema contiene un tipo de agenda que muestra que medicamentos toca a cada hora y la cantidad y si el usuario lo desea este lo alertará en forma de alarma para que no se le pase la hora que tiene programada para el suministro. Como la cantidad de información a procesar en un periodo largo de tiempo es basta, el sistema se sincroniza con un servidor para el almacenamiento de los datos y así se tenga la información disponible cuando esta sea requerida. Una opción que tiene el usuario del dispositivo móvil, es que si así lo desea, la información acerca de sus niveles de azúcar, así como información relacionada con su tratamiento la pueda mandar a un correo electrónico que será el de su médico de cabecera o su médico que la da seguimiento a su enfermedad, ya que con esto solucionamos el problema de que el paciente no puede asistir a citas por cuestiones de tiempo y como los datos que se mandaran son de las mediciones diarias de azúcar que se hace el paciente (pensamos en 3 veces al día), es completa y muestra de mejor forma el comportamiento de la enfermedad y como el cuerpo está respondiendo a los medicamentos asignados.

Implementación.

En la Figura.1. se muestra de forma general como funciona el sistema. El proceso comienza cuando el paciente, proporciona sus datos generales, patológicos, clínicos y antecedentes. Al utilizarlo por primera vez, el sistema proporciona un pre-diagnóstico y de acuerdo a este le sugiere un tratamiento, dietas y recomendaciones. Después el usuario solo ingresará sus niveles de glucosa y el sistema le proporcionará un control y administración de dichas medidas. Si el paciente llegará a presentar niveles de glucosa críticos estos serán enviados al médico de cabecera o a un familiar a través de un medio electrónico.

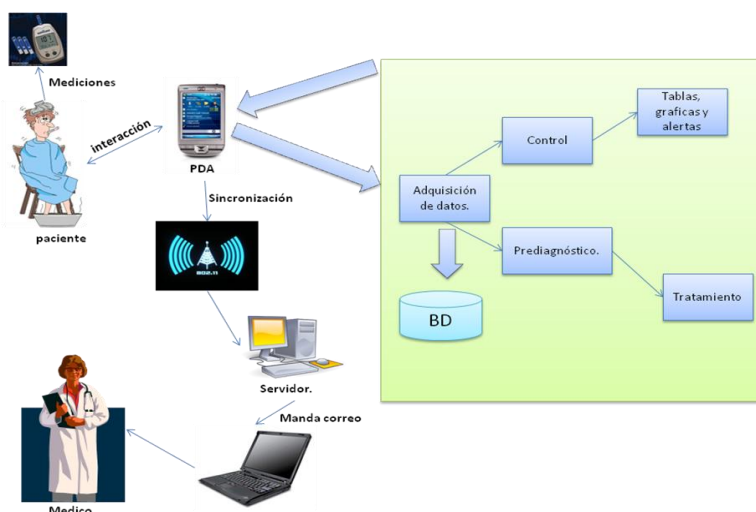


Figura 1. Esquema general del Sistema.

Módulos principales.

En los módulos que se muestran en la Figura 2, mostramos cuales son las principales funciones y que se encarga cada módulo, cabe mencionar que los módulos en donde se hace uso de lógica difusa es en pre-diagnostico, control y la generación de tratamientos, dietas y recomendaciones, ya que tenemos los resultados que nos haya arrojado el sistema experto pasamos a esos datos a otros módulos que se encargan de volver a procesar esos datos y a guardarlos.

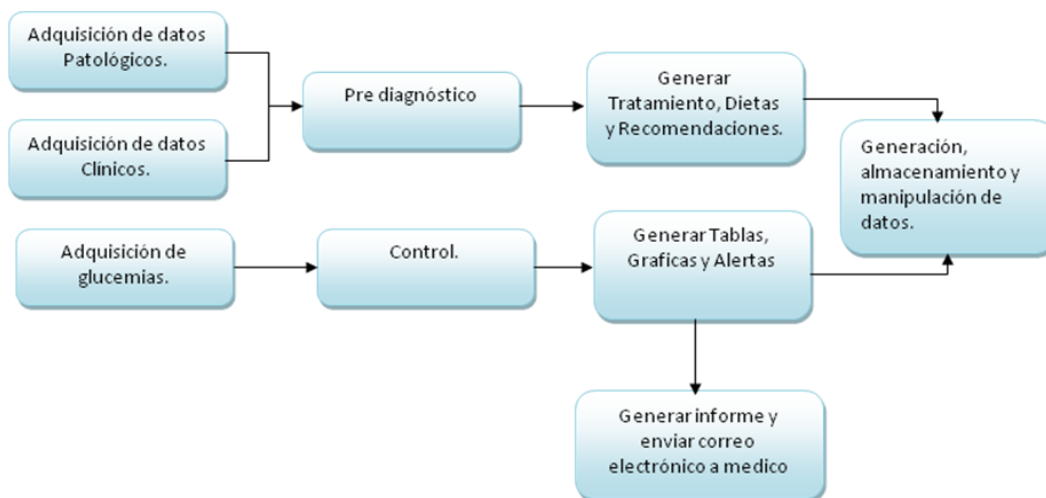


Fig. 2 Esquema general de los módulos del sistema.

Bases de la Lógica Difusa

En teoría clásica de conjuntos, un conjunto tiene unos límites nítidos (crisp). Pero, ¿cómo expresar las vaguedades inherentes a los conceptos manejados por el ser humano? Lofti Zadeh, 1965 (considerado el

Padre de la Lógica Difusa), consideró lo siguiente: La transición desde “pertenecer a un conjunto” hasta “no pertenecer a un conjunto” es gradual. [1].

Los sistemas difusos están basados en la teoría de los conjuntos difusos, las reglas difusas y los sistemas de razonamiento difusos.

En la estructura básica de los sistemas difusos se encuentran los siguientes componentes:

Base de reglas

Mecanismo de razonamiento

Universos de Entrada/Salida (E/S)

Existen tres tipos de sistemas difusos, dependiendo de la forma de las entradas y las salidas, entre estas se encuentran las siguientes:

Sistema Difuso tipo Mamdani

Sistema Difuso tipo Sugeno

Sistema Difuso tipo Tsukamoto

Fusificación.

La Fusificación es un proceso para la conversión de datos obtenidos del mundo real a un valor lingüístico lógico para lo cual se utilizan las funciones de membresía de las variables lingüísticas para calcular el grado de pertenencia para cada término en un conjunto. Este es el primer paso en difuso y debe de ser realizado para cada variable de entrada, el resultado es utilizado como entrada al mecanismo de inferencia

Para realizar la fusificación se debe de contar con las Funciones de Membresía (FM's) de las variables de entrada, estas representan gráficamente, el grado para el cual el valor real de una de ellas se traduce a un valor que pertenecen a una variable lingüística.

Las funciones de membresía más comúnmente utilizadas en el diseño de controles difusos son: la FM triangular, trapezoidal, campana y gaussiana.

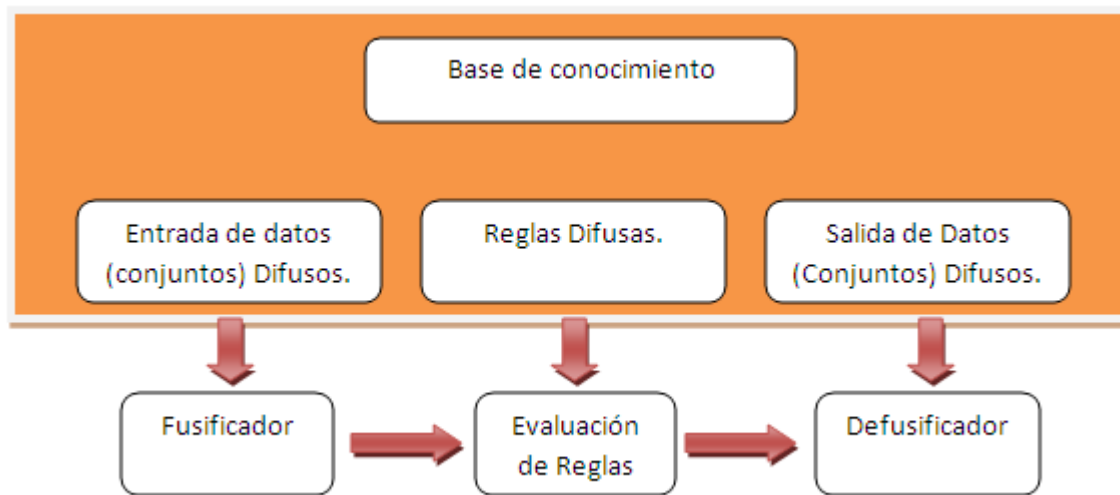


Fig. 3 Diagrama de un Sistema Difuso.

Mecanismo de inferencia.

El mecanismo de inferencia o reglas de evaluación permite interpretar y aplicar el conocimiento sobre cómo controlar mejor el sistema. Las reglas de evaluación son relaciones que se utilizan para expresar la relación existente entre los conceptos imprecisos y el comportamiento que gobierna al sistema que se quiere controlar, el experto debe desarrollar todo un conjunto de estas reglas -tantas como sean necesarias- para lograr una buena descripción del sistema a controlar. Cada regla tiene la forma de una declaración IF – THEN. La parte

IF de la regla contiene una o más condiciones, llamadas antecedentes. La parte THEN de la regla contiene una o más acciones, llamadas consecuencias. Los antecedentes de las reglas corresponden directamente al grado de membresía (entrada difusa) calculada durante el proceso de fusificación. La cantidad de reglas depende de todas las posibles combinaciones que se puedan dar entre las funciones de membresía de la entrada y la salida. Cada una de estas reglas cuentan con un grado de soporte o prioridad, el cual indica que regla contribuirá en mayor o menor parte a la salida generada por el sistema en determinado momento; la asignación de estos grados de soporte depende de la experiencia del diseñador en el proceso de control. El grado de soporte permite no ser tan radical en la evaluación de las reglas ya que permite tener una evaluación parcial de las mismas. Los antecedentes de cada una de las reglas generadas son relacionados con algún tipo de Operador difuso. Generalmente en el diseño de controles difusos son utilizados como mínimo 2 operadores: el operador AND (intersección) para el mínimo, el operador OR (unión) para el máximo; la selección del operador depende en gran parte del sistema de inferencia a utilizar.

Defusificación

El resultado final del sistema lógico difuso es la determinación de un valor de salida real que pueda ser utilizado por la aplicación o proceso de control. El valor difuso es convertido a un valor real en el proceso de Defusificación. Existen varios métodos diferentes de Defusificación utilizados en los sistemas de lógica difusa. Cada método de Defusificación utiliza una fórmula matemática para calcular la salida final basado en la fuerza de todas las reglas lógicas que han sido disparadas. Las variables en las fórmulas son grados de membresía de entrada y salida. Para realizar la Defusificación se debe de contar con las funciones de membresía de la variable de salida.

Conclusiones.

Con el análisis hecho hasta el momento nos hemos dado cuenta que nuestro sistema tiene varias ventajas sobre otros sistemas existentes ya que incluirá un prediagnosticador. Hemos observado que nuestro sistema es factible ya que los requerimientos han sido obtenidos de especialistas en este caso Médicos y personas con Diabetes Mellitus (usuarios). Se encontró que existen diversas técnicas para el modelado y desarrollo de algoritmos inteligentes el cual simulara al especialista en Diabetes Mellitus, nosotros proponemos utilizar para el modelado y desarrollo lógica difusa implementado “fuzzy logic”, sin embargo para la realización de sistemas expertos también se puede usar “prolog”. La diferencia principal y por la que se decide usar Fuzzy es que usa reglas difusas y “prolog” son simplemente reglas. Es importante señalar que la sincronización de datos se planea que sea vía inalámbrica pero cabe la posibilidad que se realice en forma alámbrica, ya que no siempre se dispone de una red inalámbrica. Se pretende que el paciente pueda sincronizar sus datos en cualquier lugar público vía Wi-Fi [4], sin embargo el alcance en costos y tiempo nos limita a probar la sincronización alámbrica. Nos hemos dado cuenta, que la modelación de los sistemas expertos se basan mucho en la experiencia de el experto al que estamos modelando y cuenta mucho la experiencia que el experto tenga y de los casos que el mismo haya tratado.

Referencias.

- [1] Chin-Teng Lin and C. S. George Lee, "Neural Fuzzy Systems", Prentice Hall, EU, 1996. 1ª Ed., 783 pp. J. S. R. Jang/C. T. Sun/E. Mizutani. "Neuro-fuzzy and soft computing. Prentice Hall, EU, 1997. 1ª Ed., 607 pp.
- [2] Manual CTO de medicina y cirugía "Endocrinología" Francisco Álvarez Blasco, CTO MEDICINA, S.L. España 2007, 408 pp.
- [3] Federación Mexicana de Diabetes (2007 Julio), "Federación Mexicana de Diabetes" http://www.fmdiabetes.com/v2/paginas/d_numeros.php
- [4] Jose A. Carballar, Wi-Fi, I Instalación, Seguridad y Aplicación, Alfaomega 2007, 319 pag.
- [5] Paul Harmon, David King. Sistemas Expertos, Aplicaciones de la inteligencia artificial en la actividad empresarial.

Apoyo en un Software Educativo para el desarrollo de competencias matemáticas en la Unidad Didáctica de Probabilidad y Estadística en el nivel Superior

Elena Fabiola Ruiz Ledesma

Instituto Politécnico Nacional

Departamento de posgrado,

Escuela Superior de Cómputo,

México, D. F.

Email: efrui@ipn.mx Tel. (55) 57296000 ext. 52041

Resumen:

En el presente trabajo se muestra una parte de los resultados del proyecto de investigación que está en proceso de desarrollo en el Instituto Politécnico Nacional, estos resultados están enfocados al desarrollo de las competencias de razonar, comunicar, resolver problemas y usar herramientas computacionales, mediante el diseño y aplicación de un software educativo de Probabilidad y Estadística, debido a que se tiene como antecedente que el 40 % de los estudiantes de una muestra de 517 alumnos que cursaban la unidad didáctica en el nivel superior, la reprobaron durante el ciclo escolar 2008-2009/I además de las dificultades que manifestaron tener en su aprendizaje. Se emplea la tecnología como una competencia en sí misma y se encuentra que los estudiantes que resuelven problemas mediante el empleo de simulaciones, logran comunicar sus ideas y emplear el software educativo como una herramienta práctica.

Abstract:

These work is a part of the results of the research project is under development at the National Polytechnic Institute, these results are focused on developing the skills to reason, communicate, solve problems and use computer tools, by design and implementation of educational software in Probability and Statistics, because it is history that 40% of students in a sample of 517 students who attended the teaching unit on the upper level, the failed during the school year 2008 - 2009 / I also reported having difficulties in their learning. Technology is used as a competition in itself and is that students solve problems by using simulations, they manage to communicate their ideas and use educational software as a practical tool.

1. Introducción

La sociedad está reclamando nuevas competencias a los profesionales y a los ciudadanos en general, que requieren el dominio de destrezas y habilidades específicas, de esta forma, surge un doble posicionamiento: formar sobre esas competencias en el ámbito profesional o desarrollarlas en el ámbito académico previo al laboral. Muchas universidades en diferentes países están rediseñando sus carreras a través de nuevos perfiles académico-profesionales en los que incluyen una serie de competencias. Las competencias son factores de superación individual y grupal que permiten el desarrollo de los recursos personales para integrarlos en las posibilidades del entorno y obtener así, de esa complementariedad, el mayor beneficio mutuo.

El Modelo educativo de IPN contempla el énfasis en una educación centrada en el aprendizaje con preferencia a una educación centrada en la enseñanza. Al contraponer enseñanza y aprendizaje se pretende resaltar la importancia que en el nuevo paradigma educativo debe tener la educación en términos de adquisición por parte del estudiante de capacidades, habilidades, competencias y valores que le permitan una progresiva actualización de los conocimientos a lo largo de toda su vida.

2. Competencias matemáticas y uso de la Tecnología

2.1 Antecedentes.

Dada la conciencia, cada día mayor y más extendida, del alto grado de provisionalidad de los conocimientos adquiridos en un momento dado de nuestra vida, parece conveniente desplazar el acento desde una educación que hasta ahora ha estado preferentemente centrada en la enseñanza de contenidos hacia una educación orientada al aprendizaje de competencias.

La educación deberá centrarse en la adquisición de competencias por parte del alumno, por lo que el papel fundamental del profesor debe ser el de ayudar al estudiante en el proceso de dicha adquisición. El concepto de competencia pone el acento en los resultados del aprendizaje, en lo que el alumno es capaz de hacer al término del proceso educativo y en los procedimientos que le permitirán continuar aprendiendo de forma autónoma a lo largo de su vida.

Investigaciones realizadas en torno a competencias PISA (OECD, 2004) [1], Rico, L. (2005), [2], Villa, A., Poblete, A (2007) [3], Sánchez s., Fortes, I. (2008) [4]; señalan que hay competencias genéricas o transversales, (instrumentales, interpersonales y sistémicas) necesarias y las competencias específicas (propias de cada profesión) con el propósito de capacitar a la persona sobre los conocimientos científicos y técnicos, su capacidad de aplicarlos en contextos diversos y complejos, integrándolos con sus propias actitudes y valores en un modo propio de actuar personal y profesionalmente. Estas competencias habían sido definidas inicialmente en el Marco Pedagógico de la UD (2001) [5] y que ahora se explicitan con tres niveles, varios indicadores y cinco descriptores para facilitar su desarrollo y evaluación.

Las competencias o procesos generales elegidos por el proyecto PISA (OECD, 2004, p. 40), son:

- pensar y razonar
- argumentar
- comunicar
- modelar
- plantear y resolver problemas
- representar
- utilizar el lenguaje simbólico, formal y técnico y las operaciones
- usar herramientas y recursos.

El presente artículo se enfoca a cuatro de las competencias señaladas: razonar, comunicar, resolver problemas y usar recursos y herramientas y se muestra cómo el diseño y aplicación de un Software Educativo para la Enseñanza de la Probabilidad y Estadística; contribuyó al desarrollo de dichas competencias. El empleo del software se justifica porque la incorporación de las nuevas tecnologías se aplica a todos los ámbitos universitarios: gestión, investigación y enseñanza.

De forma más específica, una parte importante de la innovación pedagógica se apoya en el uso didáctico y pedagógico de estas tecnologías tanto por parte del profesorado como de sus alumnos. Esta utilización ayuda a crear nuevos espacios virtuales que favorecen la autonomía de alumnos multiplican las posibilidades educativas de interacción y creación.

2.2 Sobre el uso de la tecnología

Una de las características más importantes del mundo actual es el enorme intercambio de información que se da a diario entre millones de personas, así como el desarrollo de la tecnología adecuada para transmitir con rapidez dicha información.

La presencia de las nuevas tecnologías de la información como valioso instrumento de apoyo dentro del proceso de enseñanza-aprendizaje, la interacción maestro-alumno en el aula debe continuar como eje de la actividad educativa. Deberán diseñarse modelos pedagógicos para usar eficientemente los recursos, nuevas modalidades y espacios para aprender, así como desarrollar nuevas habilidades y competencias en el estudiante.

2.3 Sobre las dificultades en la Probabilidad

Se considera que las dificultades para la correcta enseñanza de la probabilidad se deben a dos factores importantes como lo señala Ruiz E. F: (2007) [6] y Ruiz, E.F (2009) [7], derivado del proyecto de investigación con registro en la SIP 20090330. El primer factor es la falta de desarrollo de distintas competencias en los estudiantes desde los niveles previos, y que son requeridos por el estudiante competencias como razonamiento lógico, resolución de problemas, entre otras. El segundo es la actitud que toman los alumnos ante los temas de la asignatura, como por ejemplo el desinterés y la inasistencia a clases. Podría deducirse que de este último depende primordialmente la dificultad, ya que si los alumnos mostraran una actitud positiva, sería más fácil impartir la materia.

Después de un estudio realizado Ruiz E. F. (2007 y 2009). Proyecto de investigación 20090330, encontró que alrededor de un 40% de la población de los estudiantes de la Escuela Superior de Computo del IPN, no acreditan la materia, y presenta dificultades para comprender los temas señalados por la academia en el temario, por ello y debido al hecho de desarrollar la competencia del uso de la tecnología en el estudiante es que se genera un Software educativo de apoyo a los estudiantes con el contenido temático de la unidad didáctica denominada Probabilidad y Estadística.

3.0 Aspectos Teóricos

3.1 Sobre el uso de las TIC's como una competencia

El empleo de las tecnologías de la Información y Comunicación también es considerada en sí como una competencia Sánchez S., Fortes, I. (2008), [4] la cual se relaciona con la gestión de la información y de la comunicación apoyada en las amplias tecnologías a las que da acceso el ordenador personal.

En conjunto se trata de alcanzar una situación en la que la persona se desenvuelve con acierto y cierta soltura ante un ordenador personal; al menos en relación con las aplicaciones y tareas más comunes en la gran mayoría de ámbitos de utilización. La Computadora es un elemento casi omnipresente en los escenarios de trabajo académico y profesional. Cualquier estudiante o profesional se ve en la necesidad de adaptarse a este nuevo aspecto del entorno, y a los que puedan provenir en el futuro desde la misma dirección.

Por otro lado, si el alumno quiere alcanzar un alto nivel de dominio en el uso del ordenador, precisará igualmente de competencia para la innovación y la orientación al aprendizaje. Para poder obtener cierta eficacia y eficiencia no basta con limitarse a utilizar la herramienta informática a un nivel muy básico o siempre de la manera en que se realiza en un principio, sino que la competencia de innovación conduce a nuevos modos de utilización del ordenador, los cuales podrán repercutir en un mejor desempeño académico o profesional o, al menos, lo facilitarán.

La comunicación escrita también está relacionada con ella. El usuario del ordenador muestra continuamente su grado de competencia para comunicarse adecuadamente, de modo que un buen dominio técnico con una mala capacidad comunicativa da como resultado una mala utilización del medio.

4.0 Planteamiento de la Problemática

Después de un estudio realizado Ruiz E. F. (2007 y 2009). Proyecto de investigación 20090330, encontró que alrededor de un 40% de la población de los estudiantes de la Escuela Superior de Computo del IPN, no acreditan la materia, y presenta dificultades para comprender los temas señalados por la academia en el temario, por ello y debido al hecho de desarrollar la competencia del uso de la tecnología en el estudiante es que se genera un Software educativo de apoyo a los estudiantes con el contenido temático de la unidad didáctica denominada Probabilidad y Estadística.

5.0 Aspectos Metodológicos

5.1 Aplicación de una encuesta

Se aplicó una encuesta a 517 estudiantes que cursaban la unidad didáctica de Probabilidad y Estadística que forma parte del tronco común de la carrera de Ingeniería en Sistemas Computacionales en el tercer semestre y se obtuvo lo siguiente: en el primer periodo de evaluación de una muestra de 517 alumnos el 43% no lo aprobó, en el segundo departamental, observamos un aumento en este índice, pues llegó a un 46%, tomando en cuenta que el tercer periodo, es el de recuperación el porcentaje de reprobación disminuyó a 40.6% sin embargo las muestras de alumnos que no acreditan la materia es casi la mitad de la población.

En la primera oportunidad de recuperación, de 149 que realizaron el extraordinario un 38% la acreditó, sigue siendo una tasa baja de aprobación, lo que confirma la necesidad de un apoyo al profesor y al estudiante, para la enseñanza de la probabilidad, a través de un Software que tenga los tópicos vistos en la materia.

Tomando en cuenta el índice de reprobación de la Escuela Superior de Cómputo en el ciclo escolar 2008-2009/1, para la materia de Probabilidad y Estadística, se consideró importante no sólo reforzar los conocimientos del alumnado a nivel superior. Sino de desarrollar las competencias de comunicar, razonar y resolver problemas, así como usar herramientas y recursos.

5.2 Diseño del Software Educativo

En la colaboración de tres estudiantes de la carrera de Sistemas Computacionales de la ESCOM se realizaron aportaciones en formato HTML, provenientes de los apuntes e investigaciones de Ruiz E. F. (2008) [8], y los temas especificados en el temario de la academia.

La primera etapa fue el diseño en papel y electrónico del contenido intelectual, es decir, de los temas explícitos y resoluciones a problemas comunes.

Se procedió a diseñar la estructura, igualmente verificada y corregida por Ruiz E. F. y miembros del grupo, que concluyó en dejarlo separado por unidades temas teoría ejercicios resueltos ejercicios para resolver (evaluaciones) herramientas adicionales

Concluyendo la estructura se llevó a código HTML a través de diversas herramientas, como Macromedia y ahí se le agregaron imágenes, sonidos, animaciones flash, y combinaciones de colores adecuadas para los estudiantes de nivel superior

Para concluir con el diseño adecuado de este libro electrónico, a lo largo de 6 meses se le han hecho mejoras, para el beneficio del mismo, gracias a la supervisión y ejecución de diferentes alumnos.

5.3 Funcionamiento del Software

El Software educativo funciona como apoyo para la Enseñanza de la Probabilidad y Estadística a Nivel Superior el cual propone abordar los temas de la asignatura de Probabilidad y Estadística.

El Libro Electrónico se presenta por medio de un CD el cual contiene todas las indicaciones y especificaciones para el mayor aprovechamiento de este.

Para poder utilizar el software educativo se siguen los siguientes pasos:

1.-Introduce el disco en la unidad de CD-ROM.

El disco automáticamente se activa y muestra una ventana de bienvenida como se ve en la figura 1. Posteriormente aparece el Menú Principal (ver figura 2), en el cual encontraremos:

- El manual de uso: aquí encontraras toda la información necesaria de cómo usar el libro electrónico educativo.
- Libro electrónico: Aquí se encuentra el software Educativo para la enseñanza de probabilidad y estadísticas a nivel superior.
- Herramientas de diseño: Aquí se encuentra concentrada toda la información técnica del software

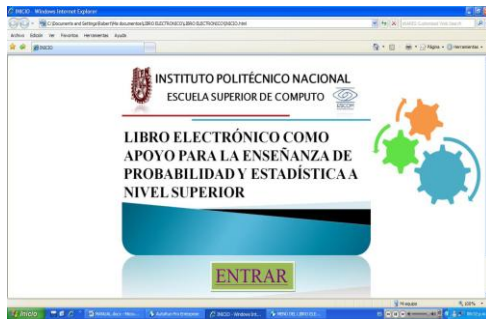


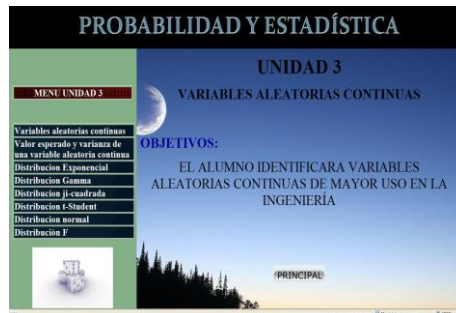
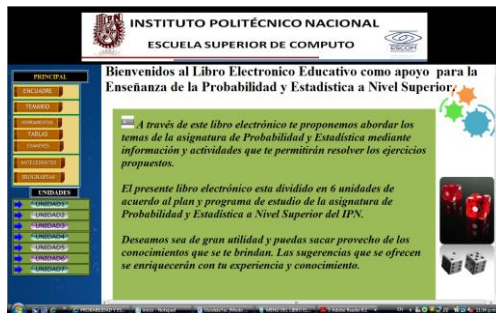
Figura. Entrada al Software Educativo



Figura 2. Menú principal

2. El software educativo se encuentra en un ambiente HTML, así que puede ser visto desde cualquier explorador web con que cuente el alumno.

Al darle clic en el botón entrar, se muestra la pantalla principal con todos los módulos contenidos en el libro. En la figura 3 se encuentra el menú el cual contiene cada uno de los aspectos básicos del curso de Probabilidad y estadística. Las primeras opciones (que se muestran con color marrón) son elementos complementarios al curso. En la pantalla principal también se puede acceder a las unidades en las cuales se divide el temario de Probabilidad y Estadística. (Ver figura 4).



Figuras 3 y 4.

Ejemplo de pantallas para acceder a los temas

Al darle doble clic al botón de alguna unidad, el alumno será re-direccionado a los temas que se ven en ésta, y podrá ir navegando en cada uno de ellos sin problema.

Cada Unidad consta de teoría, ejemplos y ejercicios propuestos, ver figuras 5 y 6, con los cuales el alumno podrá aprovechar el curso y valorar los conocimientos adquiridos con este libro educativo.

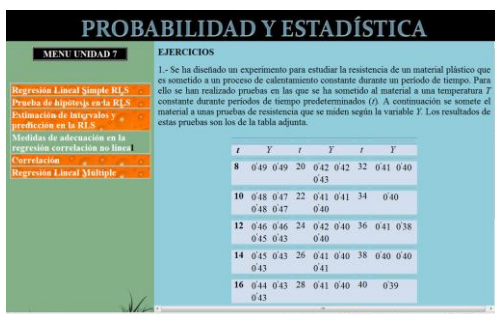


Figura 5. Pantalla como ejemplo

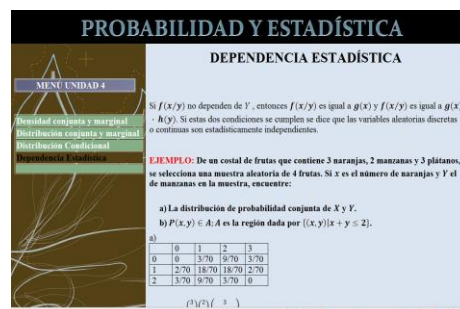


Figura 6. Ejemplo de actividad

que contiene ejercicios

5.4 Ensayo del Software

Tomando como factor para comprobar la efectividad de la creación del software educativo, se realizó un piloteo con 5 alumnos de tercer semestre de la Escuela Superior de Cómputo, a los cuales se les proporcionó una copia de evaluación del software durante todo el semestre que cursaron la materia de probabilidad y estadística.

Posteriormente el libro electrónico de probabilidad y estadística fue empleado con un grupo de 35 estudiantes durante el semestre del ciclo escolar 2008-2009/2 de la siguiente manera.

La Unidad temática de Probabilidad y Estadística está conformada por 7 unidades.

Aquí se muestra cómo se trabajó la unidad 1.

Al estudiante se le plantea el problema que dio origen a la probabilidad y para que desarrolle la competencia de razonar y resolver el problema se le pide que revise la parte de video que viene en el libro electrónico, que es un fragmento de la película cinematográfica denominada “21”, se le formulan preguntas para ayudarlo a resolver la situación.

Una vez que logra razonar lo que revisó en la película, y lo comunica a sus compañeros entonces se le pide resolver el problema planteado en un inicio. Cuando logra resolver este problema el estudiante puede avanzar en los siguientes temas de la unidad 1.

Para trabajar los problemas de combinaciones y permutaciones hay una herramienta llamada diagrama de árbol, el estudiante puede hacer uso de ésta. Los problemas cambian de valores de manera aleatoria para que no se convierta en algo mecánico al aprenderse de memoria la respuesta correcta.

Aparecen simulaciones para ayudar al estudiante a comprender la resolución de algunos problemas que se plantean.

Resultados y Conclusiones

Los primeros resultados arrojados fueron que los alumnos encontraron una manera más fácil y viable de checar alguna duda que tuvieran sobre algún tema o fórmula en específico.

Debido a que cada unidad temática traía consigo ejemplos y ejercicios, los alumnos podían ver la solución de distintos problemas y resolver alguno de los propuestos.

Mostraron mejores resultados tanto en clase como en calificaciones y esto debido a que el Software Educativo también cuenta con un apartado de exámenes por unidad, así que los estudiantes realizaron pequeños cuestionarios acerca de los temas aprendidos por unidad.

Finalmente los estudiantes externaron su agrado al software, al revisar el apartado de extras donde se encontraban todas las tablas de distribuciones de probabilidad necesarias durante todo el curso.

El software educativo por su diseño permite al estudiante realizar experimentos que en físico consume mucho tiempo y/o algunas veces no son factibles de realizarse en el salón de clases, como el lanzamiento de dados arriba de 100 veces, o el lanzamiento de una pelota a una canasta, por ello el empleo de simulaciones permite al estudiante efectuar esos experimentos y poder resolver las situaciones que se presenten.

También el software tiene programas para la obtención de distintas distribuciones de probabilidad que se requieren al resolver un problema.

El software educativo es de gran utilidad como apoyo para que el estudiante logre desarrollar competencias que en lápiz y papel llevan más tiempo.

Agradecimientos

La autora agradece el patrocinio otorgado por la Comisión y Fomento a las Actividades Académicas [COFAA-IPN] para realizar y presentar este artículo.

La investigación con número de registro 20090330 ha sido apoyada por la SIP del IPN. También agradece a sus tres estudiantes PIFI.

Referencias

- [1] OECD (2004). Learning for tomorrow's world: First results from PISA 2003. Paris: OECD.
- [2] Rico, L. (2005). La alfabetización matemática y el Proyecto PISA de la OCDE. *Padres y Madres de Alumnos. Revista de la CEAPA*, 82, 7-13.
- [3] Villa, A., Poblete, A (2007). *Aprendizaje basado en competencias una propuesta para la evaluación de las competencias genéricas*. Universidad de Deusto. Bilbao, España, p.p 59-60, 139-145, 167-177.
- [4] Sánchez s., Fortes, i. (2008). De la enseñanza al aprendizaje por competencia. Una experiencia en Matemáticas para la Ingeniería. *Actas del V Congreso Internacional de Docencia Universitaria e Innovación*. Lérída.
- [5] UD. (2001). *Marco Pedagógico UD*. Bilbao: UD
- [6] Ruiz, E. F. (2007). Estudio sobre las Nociones Básicas de Probabilidad con estudiantes de Nivel Medio Superior. *Segundo Foro de Investigación Educativa. Fomento a la Investigación Educativa.* Enero 2007. CFIE. México, D.F. IEEE Criteria for Class IE Electric Systems (Standards style), IEEE Standard 308, 1969.
- [7] Ruiz E. F. (2009) Proyecto de Investigación Métodos que evalúan competencias matemáticas en ingenierías y áreas afines. Registrado en la SIP con el número 20090330.
- [8] Ruiz, e. F. (2008). Apuntes de Probabilidad y Estadística.

Closed Form Equations for Low Order Wavelet FIR Filters

Alfonso Fernandez-Vazquez

Escuela Superior de Cómputo
 Instituto Politécnico Nacional
 Av. Juan de Dios Bátiz S/N Miguel Othón de Mendizabal
 Col. Lindavista, Del. Gustavo A. Madero
 Mexico, DF, 07738.

afernan@ieee.org

Resumen

Este artículo presenta ecuaciones de forma cerrada para filtros wavelet FIR (Finite Impulse Response). En primer lugar, muestran que el apoyo del filtro FIR es el mismo que la función wavelet correspondiente. Las ecuaciones de forma cerrada se obtienen mediante el uso de un filtro FIR auxiliares de aplanamiento máximo. Sólo tenemos en cuenta el filtro FIR baja resolución de los casos, tanto ortogonales y biorthogonales.

Abstract

This paper presents closed form equations for wavelet FIR (Finite Impulse Response) filters. We first show that the support of the FIR filter is the same as the corresponding wavelet function. The closed form equations are obtained by using a maximally flat auxiliary FIR filter. We only consider low order FIR filter for both orthogonal and biorthogonal cases.

1 Introduction

Wavelet theory is a unification theory of different ideas. For example, multiresolution signal processing used in computer vision, subband filtering techniques for speech, audio coding and image compression, and wavelet series expansions [1-4].

Most of the wavelet functions come from multirate filter banks for orthogonal case as well as biorthogonal case. In this way, the design of filter banks, which generate wavelet functions, have been proposed in the literature [2,5].

Figure 1 shows a multirate two band filter bank, where $h_0(n)$ and $h_1(n)$ are the analysis filter while $g_0(n)$ and $g_1(n)$ are the synthesis filters. Additionally, the signals $x(n)$ and $y(n)$ stand for the input and output of the system, respectively.

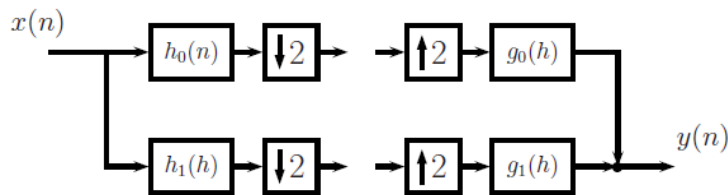


Figure 1: Multirate two band filter bank.

The corresponding z -transforms of $h_0(n)$, $h_1(n)$, $g_0(n)$ and $g_1(n)$ are denoted by $H_0(z)$, $H_1(z)$, $G_0(z)$ and $G_1(z)$.

To satisfy perfect reconstruction (PR) condition, i.e., $y(n) = x(n)$, the filters $H_0(z)$, $H_1(z)$, $G_0(z)$ and $G_1(z)$ are chosen properly, that is,

$$H_0(z)H_1(-z) + H_0(-z)H_1(z) = 1, \quad (1)$$

$$H_0(z)H_1(z) + H_0(-z)H_1(-z) = 0. \quad (2)$$

According to the multiresolution analysis, the two-scale relations are defined as follows

$$\phi(t) = \sqrt{2} \sum_n h_0(n) \phi(2t - n), \quad (3)$$

$$\psi(t) = \sqrt{2} \sum_n h_1(n) \phi(2t - n), \quad (4)$$

where $\phi(t)$ and $\psi(t)$ are, respectively, the scaling function and wavelet function. To obtain a bounded function $\phi(t)$, the filter $H_0(z)$ has as many zeros as possible at $z = -1$ [2].

Our goal in this paper is to introduce a closed form equations for low order Wavelet FIR filters $h_0(n)$ and $h_1(n)$. The approach is based on a maximally flat half band filter.

The rest of the paper is organized as follows. Section 2 demonstrates that FIR filters result in compact support wavelet. Closed form equations for low order wavelet FIR filter is discussed in Section 3.

2 Compact Support Wavelets

The case where $\phi(t)$ and $h(n)$ have compact support is very important in most applications. In this section, we show that FIR filters result in compact support wavelets.

Here we demonstrate that the supports of $\phi(t)$ and $h(n)$ are the same.

Suppose that the support of $\phi(t)$ and $h(n)$ are $[0, N-1]$ and $[N_1, N_2]$, respectively. Now by using the two-scale difference equation, we have

$$\phi(t) = \sum_{n=N_1}^{N_2} h(n) \phi(2t - n), \quad (5)$$

where the support of the right hand side is $[N_1/2, (N-1+N_2)/2]$. Because the supports in both side of the equation must be the same, we arrive at $N_1 = 0$ and $N_2 = N$.

Consequently, the design of FIR filters gives compact support basis in the multiresolution analysis.

3 Orthogonal and Biorthogonal FIR Filters

The design of PR two channel filter banks is reduced to the design of a half-band filter $S(z)$ [4], where $S(z) = H_0(z)G_0(z)$. Since we are particularly interested in the orthogonal case and linear phase biorthogonal filters, the half band filter has linear phase property and its length is $4k+3$, for k an integer [4].

Now consider the filter $S(z)$ defined as,

$$S(z) = \frac{1}{2} + \frac{1}{2} \sum_{m=0}^k s_m (z^{2m+1} + z^{-2m-1}) \quad (6)$$

The following transformation help us in the factorization of $H_0(z)$ and $F_0(z)$:

$$Z = \frac{1}{2} \left(1 - \frac{z + z^{-1}}{2} \right). \quad (7)$$

Moreover, the half-band filter is now given by

$$S_z(Z) = 2 + \sum_{m=0}^{2k+1} q_m Z^m. \quad (8)$$

Note that a simple zero of $S_z(Z)$ at $Z = 1$ corresponds a double zero of $S(z)$ at $z = -1$.

3.1 Haar wavelets

The simple case is the Haar system and is obtained from $S_z(Z) = 2(1 - Z)$. Consequently, the resulting half-band filter $S(z)$ has two zeros at $z = -1$ and $\phi(t)$ is discontinuous at $t = 0.5$.

To improve the Haar case, $S_z(z)$ must have as many zeros as possible.

3.2 Two zeros at $z = -1$

Suppose that we wish two zeros at $Z = 1$. Therefore, the filter $S_z(Z)$ becomes

$$S_z(Z) = 2(1 - Z)^2(1 + \alpha Z), \quad (9)$$

where α is a real constant. However to maintain PR condition, we have to ensure that all terms in even power of Z are zero. Therefore, the value of α is 2 and the corresponding filter $S(z)$ is

$$S(z) = 2^{-5} (1 + z^{-1})^2 (1 + z)^2 (4 - (z + z^{-1})) \quad (10)$$

Observe that the resulting filter $S(z)$ has four zeros at $z = -1$.

Now the problem is the selection of the filters $H_0(z)$ and $G_0(z)$. At first note that we can factorize $S_z(Z)$ in different ways. Particularly, the minimum phase and orthogonal filter is

$$H_0(z) = z^{-4} G_0(z^{-1}) = \frac{1 + \sqrt{3}}{4\sqrt{2}} + \frac{3 + \sqrt{3}}{4\sqrt{2}} z^{-1} + \frac{3 - \sqrt{3}}{4\sqrt{2}} z^{-2} + \frac{1 - \sqrt{3}}{4\sqrt{2}} z^{-3}, \quad (11)$$

where the number of zeros of $H_0(z)$ at $z = -1$ is two. The resulting scaling function $\phi(t)$ and wavelet function $\psi(t)$ are called Db4 (due to the length of the filter is four and was designed by Daubechies [2]) and are illustrated in Fig. 2.

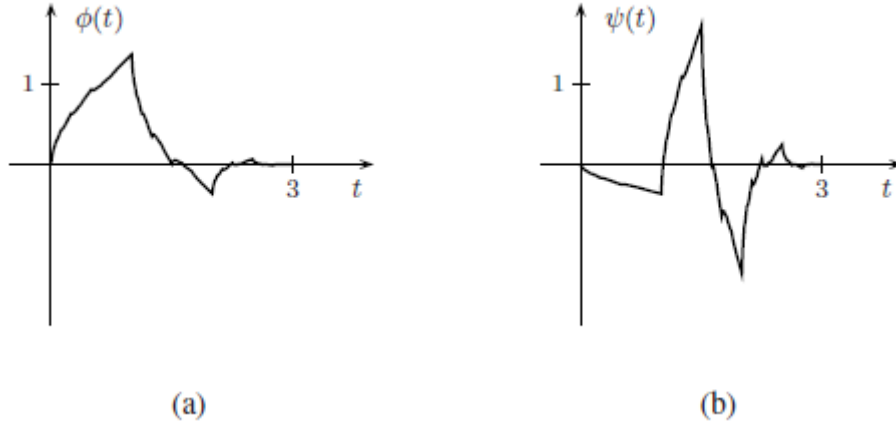


Figure 2: Scaling function and wavelet function Db4.

Another factorization is as follows

$$H_0(z) = \frac{1}{2\sqrt{2}}(1+z^{-1})^2, \quad (12)$$

$$G_0(z) = \frac{1}{4\sqrt{2}}(-1+2z^{-1}+6z^{-2}+2z^{-3}-z^{-4}). \quad (13)$$

In this case both filters have two zeros at $z = -1$. Moreover, $H_0(z)$ and $F_0(z)$ have linear phase. The set of filters is well known as *LeGall 3/5* [6]. The resulting wavelet functions are sketched in Fig. 3.

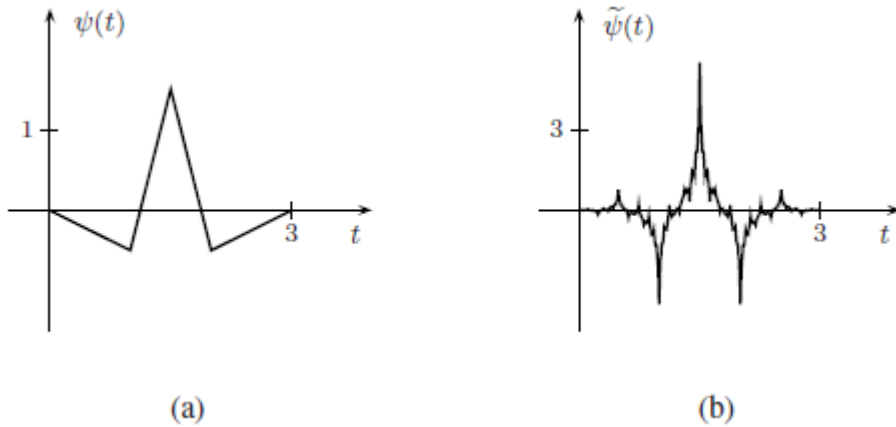


Figure 3: LeGall wavelet and dual wavelet functions.

Table 1 shows all the spectral factorization of (9), and illustrates the orthogonal case as well as the biorthogonal cases. All biorthogonal filters have linear phase.

$S_Z(Z)$	$2(1-Z)^2(1+2Z)$
Zeros of $S_Z(Z)$	$Z_{1,2} = 1$ and $Z_3 = -\frac{1}{2}$
$S(z)$	$2^{-4}(1+z^{-1})^2(1+z)^2(4-(z+z^{-1}))$
Zeros of $S(z)$	$z_{1,\dots,4} = -1$, $z_5 = 2 + \sqrt{3}$, and $z_6 = 2 - \sqrt{3}$
Daubechies filter Orthogonal	$H_0(z) = \frac{2^{-1}}{\sqrt{6-\sqrt{2}}}(1+z^{-1})^2(1-(2-\sqrt{3})z^{-1})$
Symmetric filters Biorthogonal	$G_0(z) = \sqrt{2}2^{k-5}z^{-1}(1+z^{-1})^{4-k}(4-(z+z^{-1}))$ $H_0(z) = \sqrt{2}2^{-k}(1+z^{-1})^k$, $k = 1,2,3$

Table 1: Espectral factorization of (9).

3.3 Three zeros at $z = -1$

Now, we consider the case where the filter $S_Z(Z)$ has three zeros at $Z = 1$, that is,

$$S_Z(Z) = 2(1-Z)^3(1+\alpha Z + \beta Z^2), \quad (14)$$

where α and β are real constants. To preserve the PR condition, the values of α and β are, respectively, 3 and 6.

From (14), it follows that

$$S(z) = 2^{-8}(1+z^{-1})^3(1+z)^3(38-18(z+z^{-1})+3(z^2+z^{-2})), \quad (15)$$

Accordingly, the coefficients of the orthogonal filter Db6, which give a minimum phase filter, are given by

$$\begin{aligned} 16\sqrt{2}h_0(0) &= 1 + \sqrt{10} + \sqrt{5+2\sqrt{10}}, 16\sqrt{2}h_0(1) = 5 + \sqrt{10} + 3\sqrt{5+2\sqrt{10}}, \\ 16\sqrt{2}h_0(2) &= 10 - 2\sqrt{10} + 2\sqrt{5+2\sqrt{10}}, 16\sqrt{2}h_0(3) = 10 - 2\sqrt{10} - 2\sqrt{5+2\sqrt{10}}, \\ 16\sqrt{2}h_0(4) &= 5 + \sqrt{10} - 3\sqrt{5+2\sqrt{10}}, 16\sqrt{2}h_0(5) = 1 + \sqrt{10} - \sqrt{5+2\sqrt{10}}. \end{aligned} \quad (16)$$

The corresponding filter $H_0(z)$ is 3-regular and has six coefficients, justifying the name Db6. The scaling function and wavelet, obtained from the coefficients (16), are illustrated in Fig. 4.

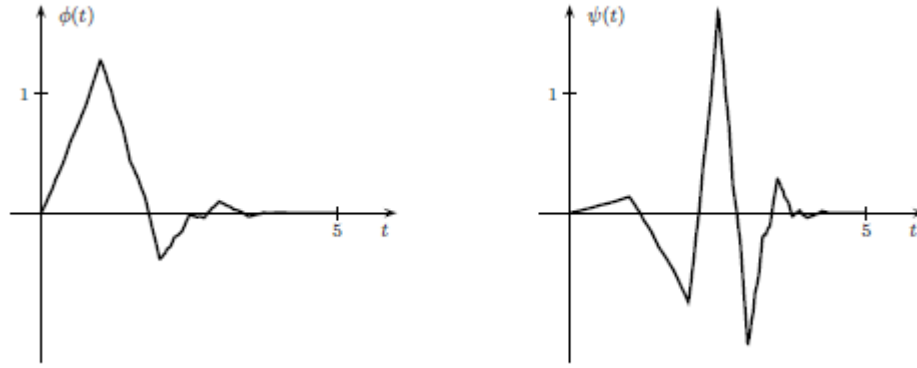


Figure 4: Scaling function and wavelet function Db6.

Table 2 shows the spectral factorizations of (14) for the orthogonal and biorthogonal case obtained from (14).

$S_z(Z)$	$2(1-Z)^3(1+3Z+6Z^2)$
Zeros of $S_z(Z)$	$Z_{1..,3} = 1, Z_4 = Z_5^* = -\frac{1}{4} + j\frac{1}{4}\sqrt{\frac{5}{3}}$
$S(z)$	$2^{-8}(1+z^{-1})^3(1+z)^3(38-18(z+z^{-1})+3(z^2+z^{-2}))$
Zeros of $S(z)$	$z_{1..,6} = -1,$ $z_7 = z_8^* = \frac{3}{2} - \frac{1}{2}\sqrt{\frac{5}{3}}\left(\sqrt{\sqrt{\frac{32}{5}}+1} + j\left(\sqrt{\sqrt{\frac{32}{5}}-1}-1\right)\right)$ $z_9 = z_{10}^* = 1/z_7 = \frac{3}{2} + \frac{1}{2}\sqrt{\frac{5}{3}}\left(\sqrt{\sqrt{\frac{32}{5}}+1} + j\left(\sqrt{\sqrt{\frac{32}{5}}-1}+1\right)\right)$
Orthogonal Daubechies filter	$H_0(z) = \frac{\sqrt{2}2^{-3}}{1+a_1+a_2}(1+z^{-1})^3(1+a_1z^{-1}+a_2z^{-2})$ $a_1 = \sqrt{\frac{5}{3}}\sqrt{\sqrt{\frac{32}{5}}+1}-3$ $a_2 = \frac{8}{3} + \frac{2}{3}\sqrt{10} - \frac{10^{1/4}}{3}\left(\sqrt{27+27\sqrt{\frac{5}{32}}} + \sqrt{5-5\sqrt{\frac{5}{32}}}\right)$
Biorthogonal symmetric filters	$G_0(z) = \sqrt{2}2^{k-9}z^{-2}(1+z^{-1})^{6-k}(38-18(z+z^{-1})+3(z^2+z^{-2}))$ $H_0(z) = \sqrt{2}2^{-k}(1+z^{-1})^k, \quad k = 1, \dots, 5$

Table 2: Spectral factorizations of (14).

3.4 Four zeros at $z = -1$

Finally, if we wish four zeros of $S(Z)$ at $Z = 1$, then the filter $S(Z)$ is given by

$$S(Z) = 2(1-Z)^4(1+\alpha Z + \beta Z^2 + \gamma Z^3), \quad (17)$$

where the values of the constants α , β , and γ are, respectively, 4, 10, and 20. Consequently, the following relation holds,

$$S(z) = 2^{-11}(1+z^{-1})^4(1+z)^4(208-131(z+z^{-1})+40(z^2+z^{-2})-5(z^3+z^{-3})). \quad (18)$$

Observe that $S(z)$ has eight zeros at $z = -1$. Besides the term $(1+z^{-1})^8$, $S(z)$ has a linear phase term, which gives six additional zeros; three zeros inside the unit circle and three outside the unit circle. This means that we can obtain two different orthogonal filters. The first one is a minimum phase filter or Db8 and the second one is a mixed phase filter, which is called *Symlet* filter [2].

Unfortunately, there are not available closed form equations to compute the filter coefficients in the orthogonal cases. Figures 5 illustrates the resulting scaling function and wavelet Db8 along with scaling function and wavelet Symlet 8.

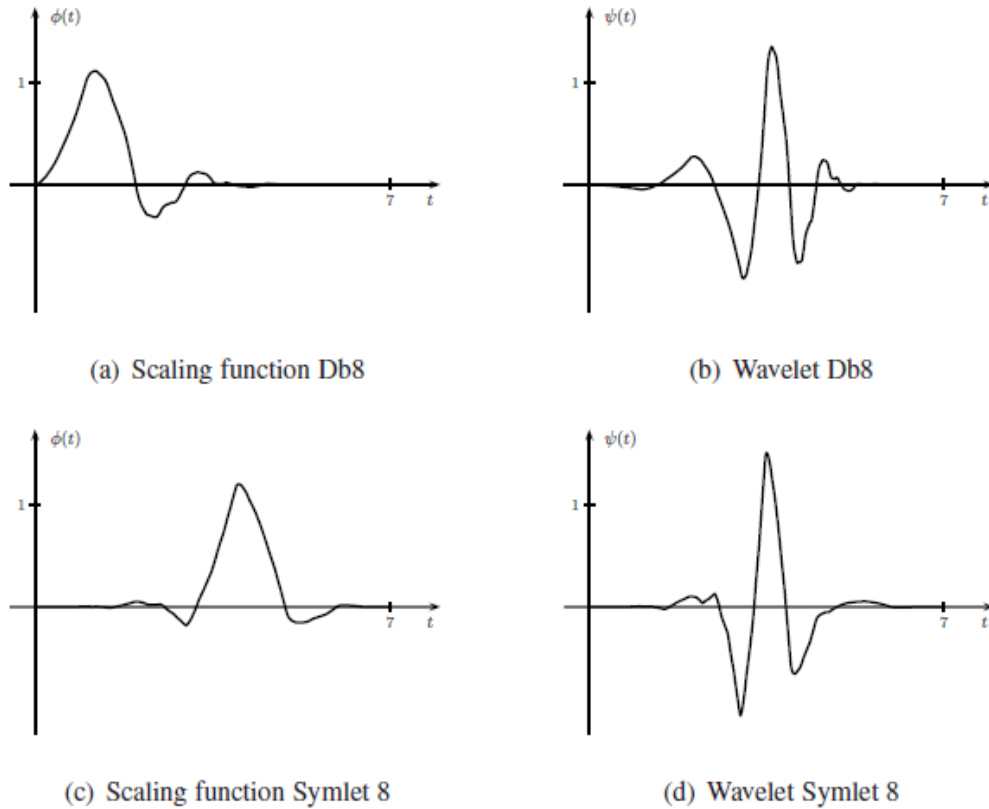


Figure 5: Scaling functions and wavelets Db8 and Symlet 8.

An interesting biorthogonal case is called *Daubechies 9/7*, where the corresponding filters $H_0(z)$ and $G_0(z)$ are given by [7]

$$H_0(z) = \frac{\sqrt{2}2^{-6}z^{-2}}{3-\rho+16/5\rho}(1+z^{-1})^4\left(2+\frac{64}{5\rho}-2(4-\rho)(z+z^{-1})+z^2+z^{-2}\right). \quad (19)$$

$$G_0(z) = \frac{\sqrt{2}2^{-5}z^{-1}}{1-\rho}(1+z^{-1})^4(z+z^{-1}-2\rho), \quad (20)$$

where

$$\rho = \frac{4}{3} + \frac{1}{3}\sqrt[3]{\frac{7}{25}\left(\sqrt[3]{10+3\sqrt{15}} - \sqrt[3]{3\sqrt{15}-10}\right)} \quad (21)$$

Like the orthogonal case, these filters have 4 zeros at $z = -1$.¹

The corresponding scaling functions and wavelets are sketched in Fig 6.

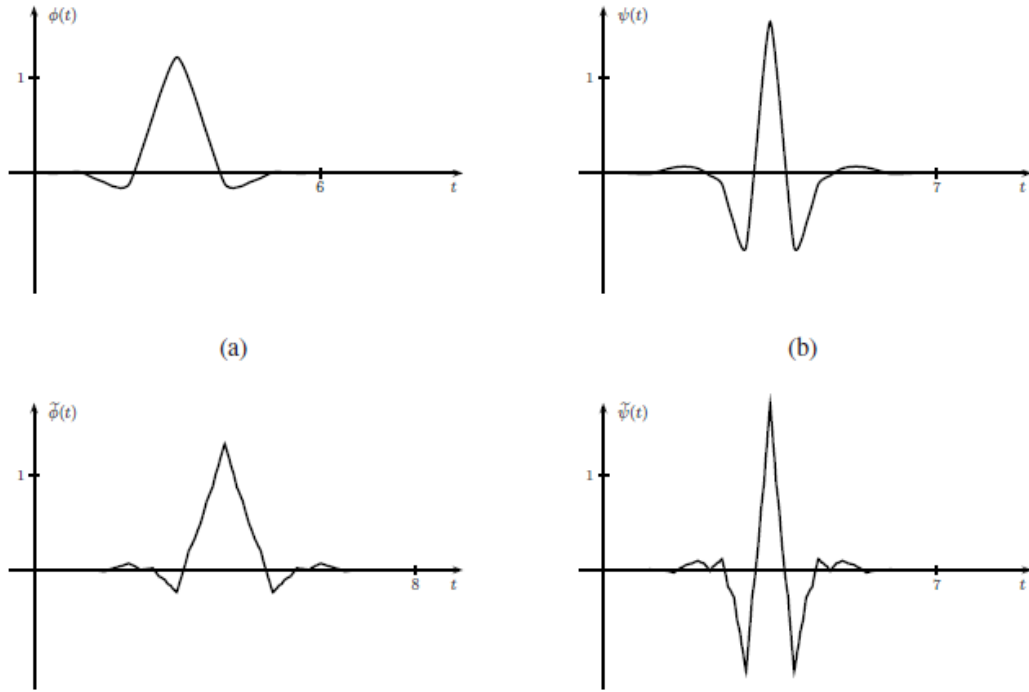


Figure 6: Daubechies 9/7 scaling functions and wavelets.

Table 3 shows others biorthogonal filters.

3.5 General case

In a similar way, we can obtain high order filters for both orthogonal (including minimum phase Daubechies filters and mixed phase Symlet filters²) and biorthogonal cases. The closed form equation for $S_z(Z)$ is [8]

¹Jpeg2000 has been approved as an international standard for the compression of still digital images. This new standard uses LeGall 5/3 as well as Daubechies 9/7 wavelet filters.

²It is worthy to highlight that to obtain the orthogonal Symlet filter the zeros of $S(z)$ are selected such that the resulting filter $H_0(z)$ is nearly symmetric, justifying the name *Symlet*.

$S_z(Z)$	$2(1-Z)^4(1+4Z+10Z^2+20Z^3)$
Zeros of $S_z(Z)$	$Z_5 = (1-\rho)/2 = -\frac{1}{6} - \frac{1}{6}\sqrt[3]{\frac{7}{25}\left(\sqrt[3]{10+3\sqrt{15}} - \sqrt[3]{3\sqrt{15}-10}\right)}$ $Z_{1,\dots,4} = 1, Z_6 = Z_7^* = -\frac{1}{6} + \frac{1}{6}\sqrt[3]{\frac{7}{25}\left(e^{-j\pi/3}\sqrt[3]{10+3\sqrt{15}} - e^{j\pi/3}\sqrt[3]{3\sqrt{15}-10}\right)}$ $= \frac{1}{2}(\rho/2-1) - j\frac{1}{2}\sqrt{\frac{20\rho^2-51\rho+48}{20\rho}}$
$S(z)$	$2^{-11}(1+z^{-1})^4(1+z)^4(208-131(z+z^{-1})+40(z^2+z^{-2})-5(z^3+z^{-3}))$
Zeros of $S(z)$	$z_{9,10} = \rho \pm \sqrt{\rho^2-1}$ $z_{1,\dots,8} = -1, z_{11} = z_{12}^* = 1-2Z_2 + \sqrt{(1-2Z_2)^2-1}$ $z_{13} = z_{14}^* = 1-2Z_3 - \sqrt{(1-2Z_3)^2-1}$
Biorthogonal symmetric filters, case I	$H_0(z) = \frac{\sqrt{2}2^{l-10}z^{-2}}{3-\rho+16/5\rho}(1+z^{-1})^{8-l}\left(2+\frac{64}{5\rho}-2(4-\rho)(z+z^{-1})+z^2+z^{-2}\right)$ $G_0(z) = \frac{\sqrt{2}2^{-l-1}z^{-1}}{1-\rho}(1+z^{-1})^l(z+z^{-1}-2\rho), \quad l=1,\dots,7$
Biorthogonal symmetric filters, case II	$H_0(z) = \sqrt{2}2^{k-12}z^{-3}(1+z^{-1})^{8-k}(208-131(z+z^{-1})+40(z^2+z^{-2})-5(z^3+z^{-3}))$ $G_0(z) = \sqrt{2}2^{-k}(1+z^{-1})^k, \quad k=1,\dots,7$

Table 3: Spectral factorization of (18).

$$S_z(Z) = 2(1-Z)^K \sum_{k=0}^{K-1} \binom{K+k-1}{k} Z^k, \quad (22)$$

where K is a positive integer, which indicates the number of zeros of the orthogonal filter $H_0(z)$ at $z = -1$, and the binomial coefficient is given by [9]

$$\binom{N}{n} = \frac{N!}{n!(N-n)!}. \quad (23)$$

Furthermore, there exists a closed form equation for the filter $S(z)$, that is,

$$S(z) = 1 + \sum_{m=0}^k s_m (z^{-2m-1} + z^{2m+1}), \quad (24)$$

where the corresponding coefficients s_m are expressed by

$$s_m = 2 \frac{(-1)^{k+m+1} \prod_{i=1}^{2(k+1)} (k-i+3/2)}{(k-m)!(k+m+1)!(2m+1)!}. \quad (25)$$

It is worth to note that the amplitude response of $S(z)$ is maximally flat at the points $\omega = 0$ and $\omega = \pi$ because it has a maximum number of zeros at $z = -1$. Figure 7 illustrates the amplitude responses of $S(z)$ for $k = 1, \dots, 4$.

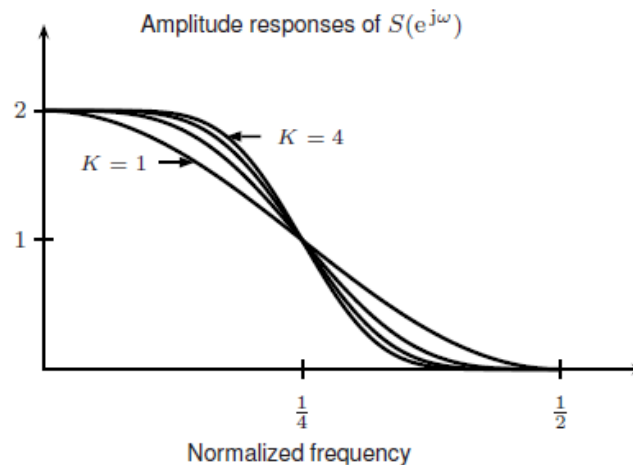


Figure 7: Amplitude response of $S(z)$ for different values of K .

4 Conclusions

We have derived closed form equation for low order wavelet FIR filter for orthogonal case as well as biorthogonal case. This approach is based on the construction of a maximally flat half band filter. The number of zeros at $z = -1$ of the corresponding half band filter considered in this paper is 1,2,3 and 4. Closed form equations for the maximally flat half band filter are also provided.

References

- [1] C. S. Burrus, R. A. Gopinath, and H. Guo, Introduction to Wavelets Transforms: A primer, Prentice Hall, 1998.
- [2] I. Daubechies, Ten Lectures on Wavelets. SIAM, Philadelphia, 1992.
- [3] S. G. Mallat, A wavelet tour of signal processing, 2nd ed. Academic Press, 1998.
- [4] P. P. Vaidyanathan, Multirate Systems and Filter Banks. Englewood Cliffs, NJ: Prentice Hall, 1993.
- [5] X. Zhang, T. Muguruma, and T. Yoshikawa, "Design of orthogonal symmetric wavelet filter using real allpass filters," Signal Processing, vol. 80, no. 8, pp. 1551-1559, August 2000.
- [6] D. LeGall and A. Tabatabai, "Subband coding of digital images using symmetric short kernel filters and arithmetic coding techniques," in Proc IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP'88), New York, 1988, pp. 761-765.
- [7] M. Unser and T. Blu, "Mathematical properties of the {JPEG}2000 wavelet filters," IEEE Trans. Image Process., vol. 12, no. 9, pp.1080-1090, September 2003.
- [8] O. Herrmann, "On the approximation problem in nonrecursive digital filter design," IEEE Trans. Circuit Theory, vol. CT--18, pp. 411--413, May 1971.
- [9] R. Ansari, C. Guillemot, and J. F. Kaiser, "Wavelet construction using Lagrange halfband filters," IEEE Trans. Circuits Syst., vol. 38, no. 9, pp. 1116-1118, September 1991.

Pruebas de un algoritmo de compresión fractal basado en quadtree para imagen

Jesús Yaljá Montiel Pérez; Rodolfo Romero Herrera, José Luis López Bonilla.

Escuela Superior de Cómputo del IPN
Av. Juan de Dios Bátiz S/N Miguel Othón de Mendizábal Col. Lindavista Del. G. A. Madero. México DF. CP 07738
yalja@ipn.mx, rromero@ipn.mx; jlopezb@ipn.mx

Resumen:

En el presente artículo se describe una técnica de compresión fractal basada en quadtree, así como pruebas fundamentales de calidad de una imagen compresada de hasta una razón de 1:60.

Abstract:

This article describes a compression technique based on quadtree fractal and fundamental tests of quality of an image to compress a ratio of 1:60.

Palabras clave: Compresión fractal de imágenes, compresión de imágenes, quadtree.

Keywords: *fractal image compression, image compression, quadtree.*

Introducción

Actualmente, los algoritmos de compresión de imágenes y video son un problema abierto. La compresión de imagen es necesaria por cuestiones de espacio físico, y transferencia remota tanto en los sistemas de cómputo como en los dispositivos móviles que poseen bajos recursos en hardware.

Para el caso de las imágenes se usan formatos con compresión JPG, GIF y PNG, entre otros. El JPG tiene una razón de compresiones de hasta 1:20 con pérdidas de información y por tanto pérdidas de la calidad de la imagen. Los formatos GIF y PNG no tienen pérdidas en las imágenes y su compresión es limitada pero adecuados para imágenes con un pequeño número de colores [1].

Una posibilidad factible de compresión de imágenes es la que se basa en la teoría de los fractales, como la de sistemas de funciones iteradas (IFS). Este tipo de algoritmo tiene una razón de proporción de hasta 1:10000 pero tiene la necesidad de que el usuario a su criterio decida valores diferentes de parámetros de una imagen a otra [2]. Existe un algoritmo totalmente automático de compresión y que se basa en la auto-similitud de los elementos de una imagen con una razón de compresión cercana a 1:60, [3].

La investigación después de los algoritmos de auto-similitud se ha enfocado principalmente en los siguientes aspectos [4]:

- Reducción de los tiempos de búsqueda de dominios. Ésta es una de las áreas más investigadas en la compresión fractal, ya que es la que es más intensiva computacionalmente. Entre las investigaciones que se ha desarrollado se encuentran la búsqueda de dominios por el vecino más cercano, la búsqueda en varias dimensiones del vecino más cercano, el descarte de dominios por medio de los valores de varianza, e incluso se ha propuesto el uso de redes neuronales para la clasificación de los dominios.

- Partición de rangos. Dado que la compresión fractal utiliza la información de los rangos para la búsqueda de los dominios, se han propuesto varios esquemas de partición, entre los más conocidos se encuentra el quadtree y el HV, sin embargo existen otros métodos como son la partición binaria, la partición adaptativa, la partición por triángulos, etc.

- Mejoramiento de la calidad de imagen y reducción de tamaño de la imagen compresada. Todos los métodos buscan de forma directa o indirecta el mejoramiento de la calidad y la disminución del tamaño de la imagen compresada, ya que es en sí son una medida de la calidad del método que se está investigando.

El presente trabajo trata los resultados de un algoritmo fractal de compresión de imagen basado en quadtree y auto-similitudes. Se analizan los resultados para diferentes tipos de imagen con características generales.

Implementación de compresor de imagen

El algoritmo de compresión de imágenes que utiliza quadtree consiste básicamente en que a una imagen se le particiona en cuadros o rangos y se buscan similitudes de formas básicas, si no se encuentran se realizan transformaciones de cada cuadro, si no se encuentran similitudes, cada cuadro se particiona nuevamente en cuadros de la mitad de tamaño que los originales y se vuelven a buscar similitudes en todos los cuadros resultantes [5, 6, 7]. Este proceso básico de búsqueda de similitudes se realiza recursivamente. La información de similitudes o dominios se almacena y se dice que una forma de la imagen se repite un número de veces, en posiciones específicas de la imagen y con transformaciones específicas y esta información es la que se almacena en el archivo comprimido, Figura 1.

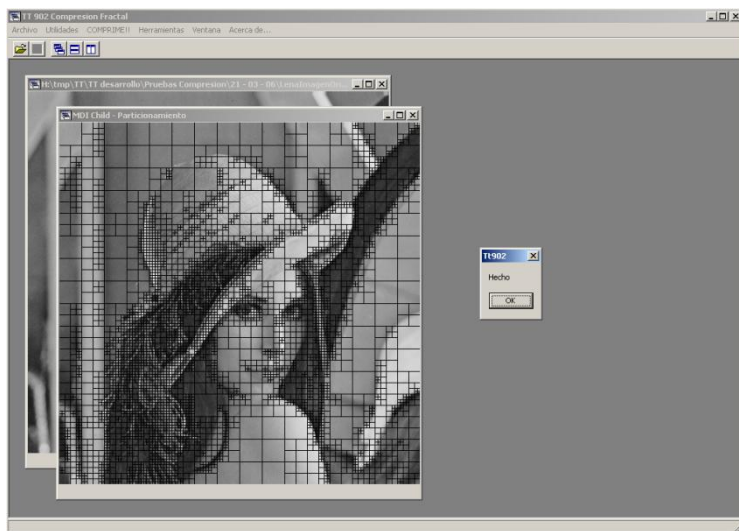


Figura 1. Imagen Lena donde se aplica las particiones o rangos con quadtree.

En la Figura 2 se muestra el diagrama de casos de uso para la compresión de una imagen. El diagrama de estados del caso de uso compresión consta de cuatro estados, los cuales son:

Estado Compresión. Este estado es el que inicia el proceso de compresión de imágenes, su único objetivo es el cargar la imagen y el inicio de la compresión en si.

Estado Partición de Rangos. Este estado tiene como función el particionamiento de la imagen y el llamado de la función de búsqueda de dominios, cuando haya terminado de particionar la imagen procederá al estado de guardado de la imagen.

Estado Búsqueda de Dominios. Este estado es el que se encargará de la búsqueda de dominio y de la obtención de la información del IFS, regresa al estado de particionamiento de rangos cuando encuentra un dominio viable.

Estado de Guardado de Imagen: Este estado es el que se encarga del guardado de la información al archivo.

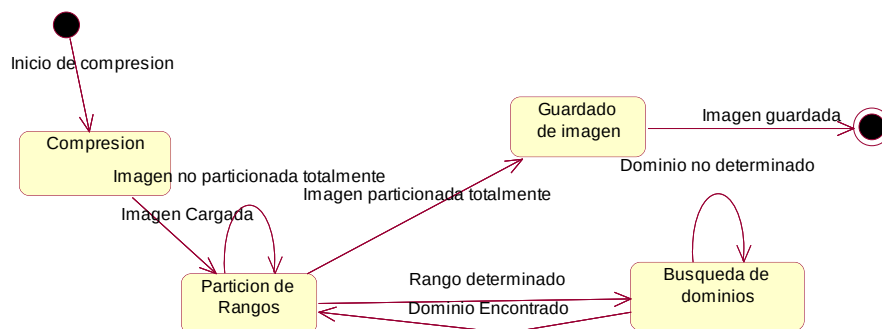


Figura 2. Diagrama de estados de caso de uso de la compresión de una imagen.

Para encontrar las similitudes o dominios en las secciones de cuadros o rangos es necesario que se realicen transformaciones contractiva de rotación, de escalamiento, y de translación. La expresión básica de estas transformaciones es [8]:

$$w \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix} \quad (1)$$

donde w es la transformación en una imagen; x, y son las coordenadas de cada pixel en una imagen, a, b, c, y d, son los valores para realizar una transformación determinada; y, e y f, son los valores para desplazamientos en los ejes x y y respectivamente. Con valores de

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} S & 0 \\ 0 & S \end{pmatrix}$$

donde S es un número real, la imagen original se escalara, reduciéndose en tamaño o aumentando su tamaño.

Para realizar una rotación se utilizan los valores de

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$$

donde θ es el ángulo de rotación de la imagen.

Para una transformación contractiva se utiliza

$$w \begin{pmatrix} x \\ y \end{pmatrix} = S \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + S \begin{pmatrix} e \\ f \end{pmatrix} \quad (2)$$

donde S es un número real de escalamiento.

En la Figura 3 se muestra el diagrama de estados de caso de uso de la descompresión que consta de tres estados:

Estado de descompresión. Este estado se encarga de la lectura del archivo y del guardado de los coeficientes de las transformaciones de dominios, después de haber guardado todas las transformaciones a memoria pasara al estado de Realizando Descompresión.

Estado Ejecutando Descompresión. Este estado efectuará las iteraciones de las transformaciones, cada transformación es un punto fijo con la cual se reconstruye la imagen.

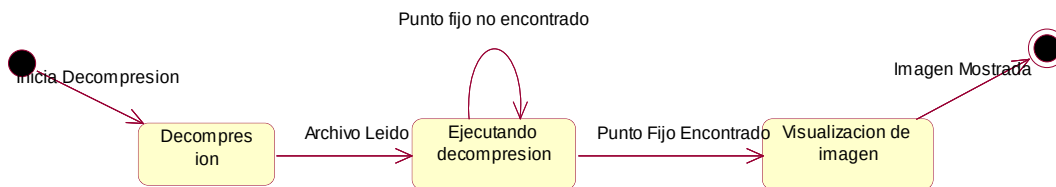


Figura 3. Diagrama de estados de uso de la descompresión de una imagen.

Con el software implementado se realizan pruebas de compresión a diferentes razones evaluando sus características y ventajas.

Pruebas y resultados

Las pruebas realizadas con el algoritmo de compresión fractal se hacen con imágenes de 512X512 pixeles en formato BMP en escala de grises con 255 niveles. Para cada imagen se realizan diferentes razones de compresión y se mide la calidad de la imagen comprimida con respecto a la imagen original.

La calidad de la imagen compresada se mide con el Peak toSignalNoise Ratio (PSNR) dado en decibel [9]:

$$MSE = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \|I(i,j) - K(i,j)\|^2 \quad (3)$$

y,

$$PSNR = 20 \cdot \log_{10} \left(\frac{MAX}{\sqrt{MSE}} \right) \quad (4)$$

donde $I(i,j)$ es el pixel i,j de la imagen original, $K(i,j)$ es el pixel i,j de la imagen compresada, M, N son las dimensiones de la imagen, MSE es el error cuadrático medio de dos imágenes, y MAX es el valor máximo de la escala de grises de un pixel, para una escala de grises es el valor de 255.

La imagen original Lena se muestra en la Figura 4.



Figura 4 Imagen original Lena en escala de grises y tamaño de 512X512 pixeles.

El algoritmo de compresión busca similitudes de dominios como se muestra en la Figura 5, estas similitudes pueden ser a un máximo detalle que es a nivel de 2X2 pixeles o en el caso mínimo de 256X256 pixeles.

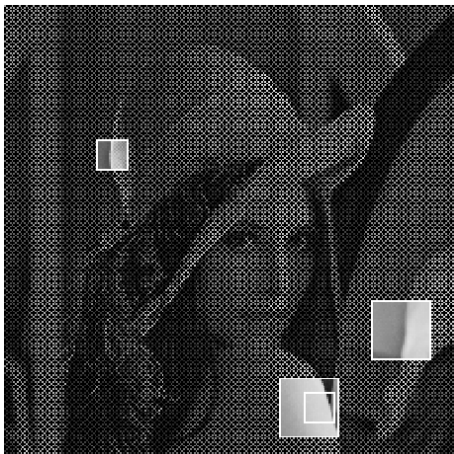
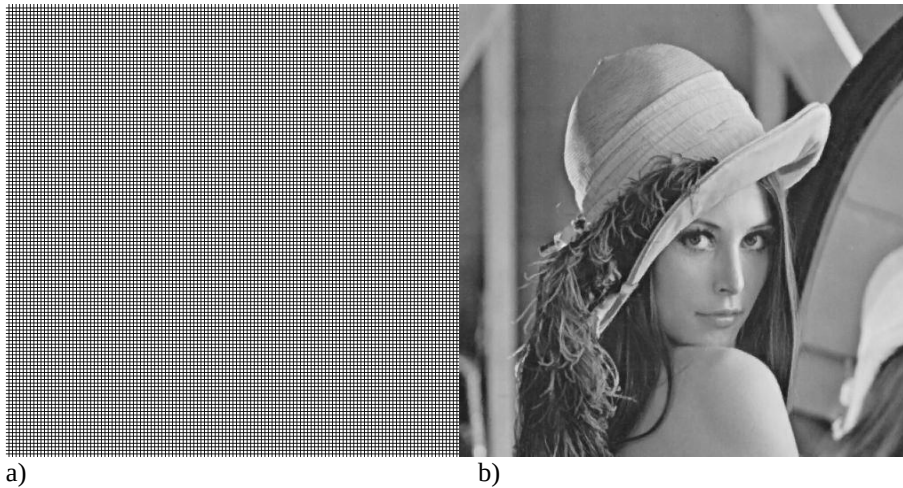
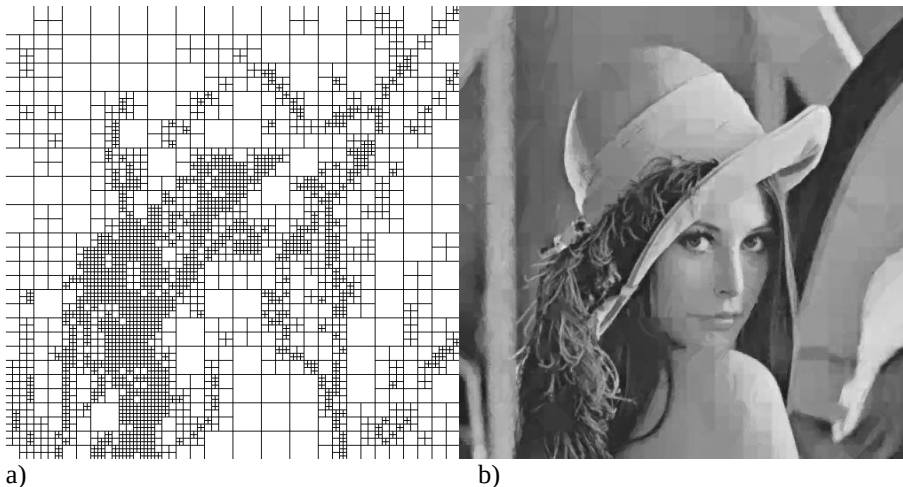


Figura 5 Imagen de Lena donde se resaltan las similitudes o dominios que se buscan para la compresión de la imagen en diferentes transformaciones.

En la Figura 6 se muestran las imágenes con la partición máxima para la imagen original Lena y la imagen descomprimida de esta. Las particiones se realizan hasta el límite posible que depende del tamaño de la imagen, en este caso se encuentran todos los dominios posibles de la imagen original. La mínima calidad para la imagen descomprimida se muestra en la Figura 7, donde en a) se muestra las particiones únicamente para las zonas donde existen variaciones de valores en los pixeles, y consecuentemente la imagen descomprimida presenta considerables pérdidas de información y que se refleja en su calidad.



a) b)
Figura 6. Imagen Lena con máxima calidad PSNR de 35.18dB y con razón de compresión de 1:14. a) máxima partición en la búsqueda de dominios y b) imagen Lena descomprimida.



a) b)
Figura 7 Imagen Lena con mínima calidad PSNR de 30.48dB y con razón de compresión de 1:61. a) mínima partición en la búsqueda de dominios y b) imagen Lena descomprimida.

En la gráfica de la Figura 8 se muestra la relación que existe entre la calidad de la imagen PSNR con respecto a la razón de compresión para el algoritmo de compresión basado en quadtree de la imagen Lena y también se incluye la compresión en un formato JPG. La imagen original contiene zonas donde no hay variación de valores en los píxeles y zonas con una gran cantidad de información de elementos como son las plumas del sombrero y pelo de la modelo.

La figura 9 muestra un aumento en la zona derecha superior del sombrero de la imagen Lena. A) muestra la descompresión de la imagen la cual fue construida a partir de dominios comunes en la imagen y b) la imagen original en formato BMP.

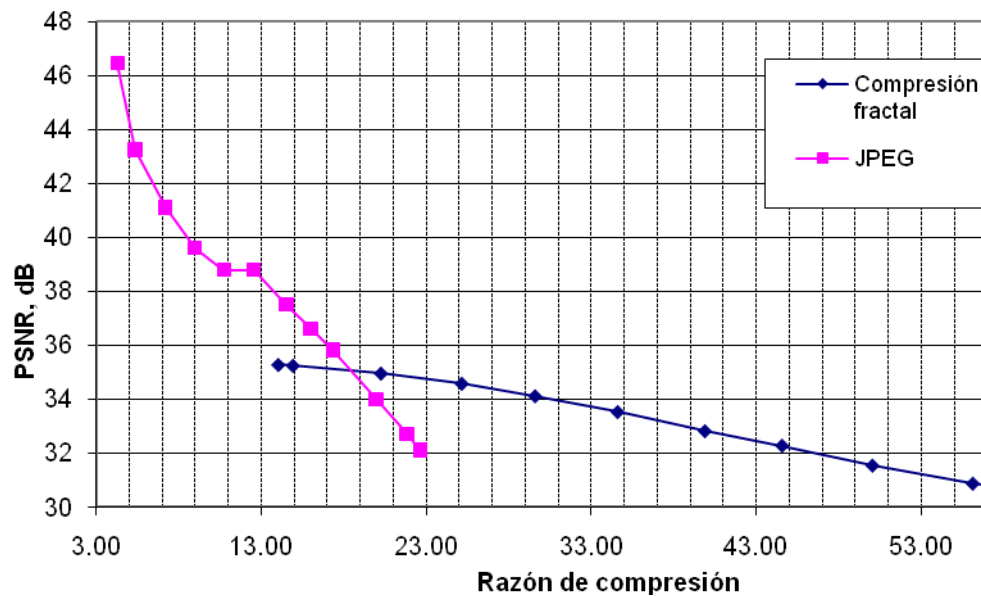


Figura 8. Calidad de la imagen Lena a diferentes razones de compresión.

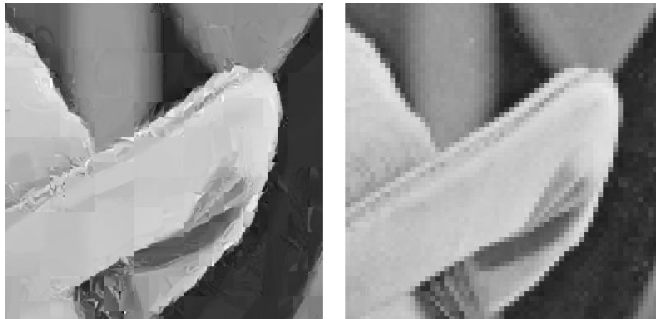
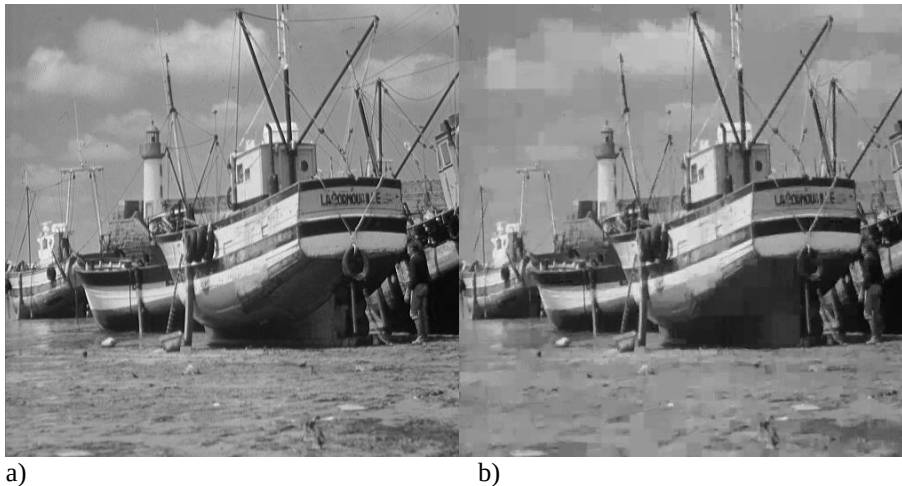


Figura 9. Acercamiento de la imagen Lena. a) acercamiento de la imagen descomprimida reconstruida a partir de dominios comunes en la imagen, y b) acercamiento de la imagen original Lena.

Una segunda prueba es con la imagen Barco en escala de grises con un tamaño de 512X512 píxeles, y en la Figura 10a se muestra. Esta imagen presenta una mayor cantidad de elementos y detalles que con la imagen Lena. La figura 10b se muestra la máxima compresión en la cual se pierden detalles y calidad de la imagen.

En la Figura 11 se muestra la gráfica de calidad de la imagen con respecto a razón de compresión, tanto para una compresión fractal como para la compresión del formato JPG.

Tanto para la Figura 8 como para la 11 se observa que el formato JPG posee una mayor calidad de imagen compresada, pero una menor razón de compresión con respecto a la compresión fractal. Por otro lado, para ambas Figuras, la razón mínima para la compresión fractal está en el intervalo para una calidad adecuada de imagen.



a) b)
 Figura 10. Imagen Bote, a) imagen original y b) imagen con máxima compresión.

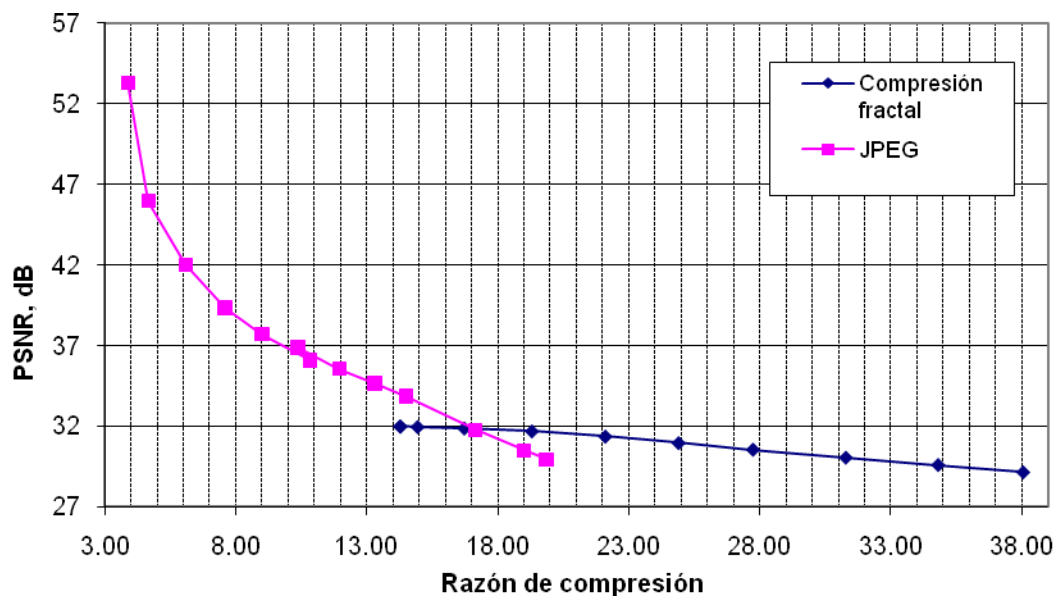


Figura 11. Calidad de la imagen Bote a diferentes razones de compresión.

Conclusiones

Los datos obtenidos de las pruebas realizadas a la compresión fractal muestran un desempeño pobre en lo que se refiere a calidad de la imagen, pero un desempeño excelente en cuanto a la compresión, ya que supera al estándar JPG. Las medidas del PSNR para la compresión fractal están dentro del intervalo aceptable para una imagen, entonces este tipo de compresión puede ser utilizado para cuando se tienen aplicaciones que se requieran archivos de imagen pequeños, como lo puede ser en las aplicaciones con dispositivos móviles.

La razón de compresión fractal de una imagen depende de las características de esta, si la imagen contiene una gran cantidad de elementos y detalles, el algoritmo puede particionarla hasta el máximo posible y esto resulta en una alta calidad de la imagen.

La imagen recuperada de la compresión fractal no se píxela al realizarle acercamientos, esto debido a que la reconstrucción se realiza con dominios de la imagen.

Las pruebas fueron realizadas con imágenes en escala de grises y para imágenes a color es el mismo procedimiento pero para cada canal del modelo de color.

Referencias

- [1] Murray JD, van Ryper W. Encyclopedia of Graphics File Formats; 2nd ed. O'Reilly & Associates Inc. 1996
- [2] W. Metzler; "Image compression predicated on recurrent iterated function Systems; 2nd International Conference on Mathematics & Statistics", June, 2008, Athens, Greece
- [3] Utku Salihoglu; Chaos in small Recurrent Neural Networks : theoretical and practical Studies; UNIVERSITÉ LIBRE DE BRUXELLES FACULTÉ DES SCIENCES DÉPARTEMENT D'INFORMATIQUE; Tesis, 2004.
- [4] Lih-Yang Wang; Ching-Hui Lai; Kuan-Ren Pan; Quad-tree-based image shape coding with block compensation; Information Technology and Applications, 2005. ICITA 2005. Third International Conference on Volume 1, Issue , 4-7 July 2005 Page(s): 716 - 719 vol.1
- [5] Eli Shusterman and Meir Feder; Image Compression via Improved Quadtree Decomposition Algorithms; IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 3, NO. 2, MARCH 1994.
- [6] Yuval Fisher Editor; "Fractal Image Compression: Theory and application"; Springer- Verlag 1995.
- [7] Ghim-Hwee Ong, Chong-Meng Chew, Yi Cao; "A simple Partitioning Approach to fractal Image Compression", 2001.
- [8] Unser, M. Neimark, M.A. Lee, C.; Affine transformations of images: a least squares formulation; Image Processing, 1994. Proceedings. ICIP-94., IEEE International Conference; 13-16 Nov 1994; Volume: 3, On page(s): 558-561 vol.3.
- [9] Linfeng Guo; PSNR-BASED OPTIMIZATION OF JPEG BASELINE COMPRESSION ON COLOR IMAGES; IEEE ICIP 2006.

Instrucciones para los autores

Los artículos que se someten a **RISCE** deben contener resultados inéditos y originales, no haber sido publicados con anterioridad ni haber sido sometidos simultáneamente a otra revista científica. Si el artículo ha sido presentado, sometido o publicado en alguna otra parte, deberá informarse al coordinador editorial. Los artículos deben ajustarse a las siguientes especificaciones:

- Idioma Inglés (anexar un resumen y palabras clave en español)
- Idioma Español (anexar un resumen y palabras clave en Inglés)
- Procesador de texto admitido: MS-Word.
- Tamaño de página: carta, utilizar un solo lado de la hoja. Máximo 10 páginas.
- Márgenes: izquierdo 2.5 cm y derecho 2 cm., superior 2.5 cm e inferior 2.5 cm.
- Autores: primer nombre seguido de los dos apellidos (sin abreviaturas), abajo: afiliación y e-mail.
- Tipo de letra del texto regular: Times o Times New Roman de 10 pt (título original 22 pt; secciones 11.5 pt, subsecciones 11.5 pt, en negritas).
- Texto: a una columna y con espaciado sencillo (renglón seguido).
- Resumen/Abstract: entre 70 y 150 palabras, colocado al principio del texto, seguido del de Español o inglés según sea el caso.
- Palabras clave/Keywords: colocadas después del resumen en negritas, y no más de 10.
- Imágenes y fotografías: deben ser de alta calidad, con colores bien definidos y contrastantes, en mapa de bits (no sectorizadas) en formato JPG e incrustadas en el texto de forma que se puedan manipular independiente.
- Fórmulas: Deberán de presentarse en formato de tabla sin bordes, centradas y la numeración de c/u justificada a la derecha con negritas en mapa de bits, no vectorizadas.
- Pies de figura. Deben mencionarse dentro del texto y numerarse de manera consecutiva con un tipo de letra Times New Roman 9 puntos
- Cabecera de tabla. Deberá presentarse en la parte superior de la tabla un numeración consecutiva y descripción con tipo de letra Times New Roman 9
- Referencias:

En cualquier caso el nombre del autor del artículo o publicación web deberá mostrarse al principio. Deberán ordenarse conforme aparezcan dentro del texto encerradas entre paréntesis cuadrado —[]—. A continuación algunos ejemplos:

- [1]. Baldonado, M., Chang, C.-C.K., Gravano, L., Paepcke, A.: The Stanford Digital Library Metadata Architecture. *Int. J. Digit. Libr.* 1 (1997) 108–121
- [2+]. Bruce, K.B., Cardelli, L., Pierce, B.C.: Comparing Object Encodings. In: Abadi, M., Ito, T. (eds.): *Theoretical Aspects of Computer Software. Lecture Notes in Computer Science*, Vol. 1281. Springer-Verlag, Berlin Heidelberg New York (1997) 415–438
- [3]. van Leeuwen, J. (ed.): *Computer Science Today. Recent Trends and Developments. Lecture Notes in Computer Science*, Vol. 1000. Springer-Verlag, Berlin Heidelberg New York (1995)
- [4]. Michalewicz, Z.: *Genetic Algorithms + Data Structures = Evolution Programs*. 3rd edn. Springer-Verlag, Berlin Heidelberg New York (1996)

Instrucciones:

Enviar el archivo en extenso a la siguiente dirección electrónica: ebustosf@gmail.com

Los revisores técnicos le harán llegar sus observaciones y modificaciones, las cuales deberá realizar y reenviar el archivo corregido al correo arriba mencionado.

El comité editorial se comunicará mediante correo electrónico indicándole la aceptación o rechazo del artículo.

Se le solicitará autorización para publicación; en caso de aceptar se le indica la cuenta donde debe hacer el depósito por cobro de publicación y el costo, el cual no debe exceder de \$1000.00 pesos mexicanos.

Reserva de Derechos 04-2008-062613190500-203