



VIII Congreso de la Sociedad Peruana de
Computación

Proceedings, CSPC-2009

25-27 de Noviembre, 2009

Huancayo, Perú

Editor: Alex J. Cuadros-Vargas



**SOCIEDAD
PERUANA DE
COMPUTACIÓN**

VIII Congreso de la Sociedad Peruana de Computación

Sociedad Peruana de Computación
Coo. La Cantuta D-11
Jose Luis Bustamante y Rivero
Arequipa-Perú

Editor: Alex J. Cuadros Vargas (alex@spc.org.pe)
Primera edición, 2011

Hecho el Depósito Legal en la Biblioteca Nacional del Perú N°: 2011-09641

ISBN: 978-9972-9876-7-0



Impreso en:
Editora Perú
Jr. Sinchi Roca 347, Nro 347
0(61) 51-01-61
ediciones_huanuco@hotmail.com
RUC : 10224992845
Agosto 2011, Huanuco-Perú
Tiraje: 50 ejemplares

Presentación

La Jornada Peruana de Computación (JPC) se llevó a cabo en la ciudad de Huancayo-Perú, del 25 al 27 de Noviembre del 2009. En su VIII versión, la JPC fue organizada conjuntamente por la Sociedad Peruana de Computación (SPC) y la Universidad Continental (UC). En los últimos años la JPC ha venido constituyéndose como la reunión más importante de académicos y profesionales de computación del país abriendo espacios para la presentación de avances científicos y tecnológicos en el área, así como el intercambio de experiencias entre sus participantes. En ésta VIII edición la JPC forma parte de un evento de mayor envergadura denominado Congreso de la Sociedad Peruana de Computación (CSPC).

La SPC invita a las comunidades académica y profesional a la presentación de trabajos en el marco de las áreas de interés del evento. La JPC es el lugar ideal para el envío de trabajos originales de iniciación científica así como aquellos relacionados con proyectos de investigación en instituciones de enseñanza superior peruanas. De la misma manera, se esperan artículos relacionados con experiencias en el ámbito tecnológico/profesional.

Marco Alvarez
Presidente del Comité Organizador
CSPC - Noviembre 2009

Organización

Congreso de la Sociedad Peruana de Computación

Marco Alvarez, USU, EEUU (Presidente)

Gabriela Jurado, UContinental, Perú (Organización local)

Comité de Programa - Jornada Peruana de Computación

Alex J. Cuadros Vargas (Peru) [chair]
Abraham Dávila (Peru)
Ana Maria Cuadros Valdivia (Peru)
Andrew Pletch (USA)
Angel G. Coca Balta (Brazil)
Antonio Castelo Filho (Brazil)
Arnulfo Alanis Garza (Mexico)
Arturo Henry Torres Zenteno (Spain)
Carlos Raymundo (Peru)
Carlos Estombelo-Montesco (Brazil)
Cesar Collazos (Colombia)
Christian Danniell Paz Trillo (Brazil)
Cristian Lopez del Alamo (Peru)
Dennis Barrios Aranibar (Peru)
Eduardo Rafael Llapa Rodríguez (Brazil)
Ernesto Cuadros-Vargas (Peru)
Giovani Librelotto (Brazil)
Graciela Lecireth Meza Lovon (Canada)
Helio Pedrini (Brazil)
Helton Biscaro (Brazil)
Herbert Luque Peralta (Brazil)
Javier Montoya (Brazil)
James Gee (USA)
João José Neto (Brazil)
Johannes Textor (Germany)
Jorge Poco (Brazil)
José Eduardo Ochoa Luna (Brazil)
Juan Carlos Gutierrez Caceres (Peru)
Juan Manuel Gutiérrez Cárdenas (Peru)
Karina Valdivia Delgado (Brazil)
Liliana Paola Mamani Sánchez (Ireland)
Lucio Andre Castro Jorge (Brazil)
Marco Alvarez (USA)
Markus Mock (USA)
Meliton Apaza (Brazil)
Nelly Condori-Fernández (Spain)
Nicolás Antezana Abarca (Peru)
Omar U. Florez (Peru)
Raquel Esperanza Patino Escarcina (Peru)

Rosane Minghim (Brazil)
Ruben Gomez Diaz (Brazil)
V́ctor Laguna (Brazil)
Yvan Tupac Valdivia (Brazil)

Edici3n de proceedings

Alex J. Cuadros Vargas (Perú)
Lucía Romina Otazú Conde (Perú)

Índice general

<i>A formal specification of spatio-temporal system in Object-Z</i> C. Vidal, L. Jiménez, P. Arce y D. Alegría	1
<i>Algoritmo voraz para emparejamiento máximo de grafos</i> L. A. Lujan, E. Rodríguez y H. Y. Sánchez	11
<i>Comparación de dos algoritmos metaheurísticos para el problema de optimización de portafolios de inversión</i> P. Fonseca, M. Luna y J. Trelles	21
<i>Evaluación colaborativa en grupos de trabajo y aprendizaje usando el framework IssueNet</i> T. Escovedo y R. N. Melo	31
<i>Extracción de características en el procesamiento digital de una señal para el mejoramiento del reconocimiento automático de habla usando Wavelets</i> J. L. Guevara y J. O. Salazar	39
<i>Implementación de servidor XS para despliegue de proyecto OLPC en escuelas del Perú</i> G. M. Chevarría, L. Atoche y W. Grández	49
<i>Modelación formal espacio-temporal: Uso de cálculo de eventos basado en objetos</i> C. Vidal, E. Vidal-Duarte, A. Zuñiga y L. Albornoz	57
<i>Planificación y seguimiento de la trayectoria de un robot móvil dentro de un ambiente controlado</i> L. Jave, S. Arauco, K. Cervantes y J. Colán	67
<i>Simulación de cirugía plástica</i> J. F. Arteaga, S. F. Obando y J. Rocca	77

A Formal Specification of Spatio-Temporal System in Object-Z

Cristian Vidal¹, Leoncio Jiménez¹, Paul Arce², David Alegría¹

¹ Business Informatics Administration, Universidad de Talca, Talca – Chile

{cvidal, lejimenez, dalegría}@utalca.cl

² IBM Global Services, Edmonton, Alberta – Canada

parce@ca.ibm.com

Abstract. The formal modelling of software systems is a non traditional task in the implementation of software applications in general. The formal modelling of high cost systems (critical systems), is a task that has had light relevancy in the last years. A Spatio-Temporal information system is a critical system, due to the high cost of the information that these systems handles. This work presents the formal modelling of a spatio-temporal information system, using a traditional modelling approach and a current object oriented formalism, Object-Z.

Keywords: Modelling, Spatio-Temporal, States, Events, Object-Z.

1 Introduction

The formal modelling and formal requirements specification of a spatio-temporal or temporal geospatial systems, is a non traditional task in software engineering, computer sciences and geoinformatics. In (Vidal, 2005) a work that consists of a formal modelling of a spatio-temporal system work is presented, in a logic deductive database system context, using the *Object Event Calculus* (OEC) formalism (Kesim, 1996).

In the modelling of a temporal geospatial system, the following modelling approaches are used: Snapshot-based, State-based and Event-based approach.

The goal of this work is to present one formal specification requirements of spatio-temporal systems, for its future implementation, using a spatio-temporal modelling approach. In this formal modelling, a formal object oriented requirement specification language is used (Object-Z), and the previously mentioned spatio-temporal modelling approaches state-base and event-base are used. Object-Z is a formal language, relatively current (Smith, 1992). That is an extension of the formal language Z (Davies, 1999). Object-Z allows the inclusion of object oriented and temporal elements modelling features, not present in the formal language Z.

This work is divided as following: in *Section 2*, the different conceptual modelling approaches of a spatio-temporal system are presented, in *Section 3*, the Object-Z formalism used in the formal modelling of spatio-temporal systems proposed in this work is presented and globally described, in *Section 4*, the Object-Z modelling or formal specification of a spatio-temporal systems is described, given the modelling approaches described in *Section 2*, in *Section 5*, the main conclusions and future works are presented.

2 Spatio-Temporal Modelling

Since it was mentioned, there are three approaches of modelling of spatio-temporary information: Snapshot-based, State-based and Event-based approach. The Snapshot-based approach is described and analyzed in (Armstrong, 1988) (Worboys, 1994) (Petry, 2003) (Vidal, 2007) (Hayes, 1987). In this work, other two approaches are in use. In the sections 2.1 and 2.2, are described these approaches.

2.1 State-based Model

The state-based model introduces the idea of timestamp to the set of objects in study. In a model based on states of the objects, the idea is to have a record of all the temporal states of an object. The state of an object is completely defined according to the value of its attributes. So, when an object changes the value of any of its attributes, then that object experiments a change of state.

The traditional state-based model consists in the utilization of a timestamp associated to each change of state of each of the objects present in the system model. This way each object of the system has associated a set of changes of state for each attribute that is affected by a change of its value. The initial value of each attribute is registered as a change of state of the object. For instance, if there is a *People class* with an identity attribute *Person*, and attributes *Attribute* and *State*. An example of the registration of changes of state associated to an object of this class (Person: Cristian), within the attributes previously mentioned, is the following (Vidal, 2007):

Table 1. Example of the classic state-based model relation Change_State.

Class People - Object Cristian

Person	Attribute	State	Vs
Cristian	Job	Programmer	01/01/2000
Cristian	Job		01/03/2000
Cristian	Job	Programmer	01/01/2001
Cristian	Job		07/03/2001
Cristian	Job	Engineer	01/08/2005
Cristian	Location	Serviu	01/01/2000
Cristian	Location		01/03/2000
Cristian	Location	Serviu	01/01/2001
Cristian	Location		07/03/2001
Cristian	Location	CBB	01/08/2005

A model very used in practical implementations of temporal databases uses a timestamp based in intervals of time, for the transaction time as for the valid time (Ts-Te and Vs-Ve, respectively) (Chen, 1989). These timestamps can be associated to any dynamic relation in which the objects participate. For instance (Vidal, 2007), if we think in a particular person like ‘Cristian’, and the jobs where this person had worked at, then this dynamic relation (Works At) could have an associated timestamp. Table 1, shows the example of an instance of the dynamic relation Works_At, where a time interval is used to represent the valid time of each tuple in the relation. It’s worth noting that the temporal granularity for all the tuples in the relation is daily.

Table 2. Example of the relation Works_At in a state-based model.

Persona	Empleo	Lugar	V _s - V _e
Cristian	Programador	Serviu	01 de Enero 2000 - 01 de Marzo de 2000
Cristian	Programador	Serviu	01 de Enero 2001 - 07 de Marzo 2001
Cristian	Ingeniero	Cementos Bio Bio.	01 de Julio 2005 - NOW

In the previous example the utilization of the value **NOW** for one side of the time interval is shown, which allows to represent a valid interval open on its right side (or of current validity) (Gregersen, 1999). When responding to a query that includes attributes with a **NOW** value, this value should be replaced by the current time value.

In general, the state-based model answers a wide range of spatio-temporal queries. Nonetheless, in its conception, the model leaves out an important set of information regarding the phenomenon’s that cause the change of states of an object. This model only focuses in the states of the objects, but it doesn’t say anything about the events that affect the states of the objects in study.

2.2 Event-based Model

A model based only on events allows the registration of spatio-temporal events occurrences, from these occurrences; the temporal state of any object is obtained. At a conceptual level, an event-based model; models the *Participation* and *Involvement* relationships between objects and events (Grenin and Smith, 2004) (Grenin, 2002). At a logical and physical level, an event-based model obtains the state of the objects, in the style of the Object-based Event Calculus or OEC (Vidal, 2005) (Kesim, 1996) (Petry, 2003).

A model based on events is a complete model, because it registers all occurrences of spatio-temporal events. In this type of model usually the set of events is restricted according to the information desired to emphasize in the problems domain. This way, the loading of unnecessary and irrelevant information can be avoided, but to obtain the state of an object, usually is a task with a high computational cost.

Although this model makes it possible to respond completely too all spatio-temporal queries, it has been empirically confirmed that this model has a high computational cost obtaining the spatio-temporal states of objects. This is mainly

because of the non existence of a state base of each object in study. This obligates the revision of a high number of tuples to be able to answer basic spatio-temporal queries (Vidal, 2007).

The literature review emphasizes the following types of events associated to objects with spatial variability (Vidal, 2007):

- **Create:** An event that creates an object.
- **Change:** An event that changes a property of an object without changing the objects identity. It is distinguished between attribute changes (internal change) and membership change (external change).
- **Destroy:** An event that destroys an object.
- **Split:** An event that creates one or more new objects from an existing object, which suffers a change on his spatial limits (and possibly in some non spatial attributes) caused by the occurrence of this event.
- **Merge:** An event that can result in the creation of a new object, due to a fusion of two or more objects which disappear after, or due to a change in the limits of an existing object (and possibly to some non spatial attributes), caused by the fusion of this existing object with one or more objects which disappear.

3 Object-Z

Object-Z is a formal requirements specification language of object oriented software, being an extension of the known language Z (Smith, 1992) (Hayes, 1987) (Spivey, 1992) (Duke, 1995) with the incorporation of object oriented concepts (Meyer, 1998). Object-Z is a current formal object oriented language, well documented and suitable for this work purpose. Z is a requirements specification model-oriented language with features for modelling complex data structures and their operations. Object-Z improves the clarity and power of modelling Z specifications, due to its object oriented features (encapsulation, inheritance and polymorphism) and temporal modelling features.

Object-Z, just like formal language Z, is a model-oriented language and its main root is the set theory; its most important feature is the class diagram. In Object-Z, a class diagram has the shape of a box with a Z style name, and optional generic parameters; including an optional section for temporal invariants of the class. *Figure 1* (Smith, 1992) (Grenin, 2002) (Hayes, 1987) illustrates the shape of an Object Z diagram class box.

The box components are:

- A visibility list that restricts access to attributes and operations;
- A class list from which its attributes and methods are inherited (superclasses);
- A definition list of the classes types and constants;
- A state diagram where the class state attributes and invariants are defined.
- An initial state diagram which specifies the initial state of the class objects.
- A set of operation diagrams; where in each one of them the pre-conditions and post-conditions of each operation of the class are specified.

- A temporal invariant or history invariant that restricts the operations order by using temporal logic.

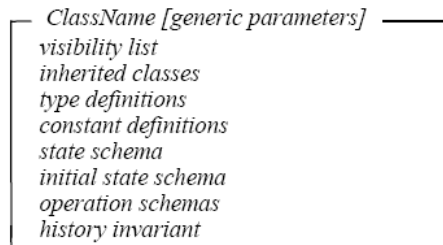


Fig. 1. Object-Z Scheme Class.

The class diagram extends the types definitions of language Z, allowing the classes to be able to act like types.

As mentioned before, the temporal or history invariant of an Object-Z class diagram makes possible to define a temporal order on the possible occurrence of the class methods. The operators $\square P Q$ and $\sqcup P Q$ are used, where P is a method of the class; indicating that the method P is always in condition Q , or eventually in condition Q respectively, where the condition Q could be **Enabled** or **Occurs** (Smith, 1992).

The use of one formal specification requirements is necessary for the implementation of complex software applications (Holloway, 1997), like a spatio-temporal information software application. In this work the modelling of a spatio-temporal information system is presented, using the modelling approaches described in *section 2*.

4 Object-Z State + Event Spatio-Temporal Modelling

One approach of modelling spatio-tempora system with Object-Z, using the approach of modelling based on states of an object, together with the record of the event that originates the change of state, is proposed.

The approach based on an objects states, is highly used in the implementation of real systems, given the natural vision of performing a registration of all the possible changes of an object. The extension presented in this section, is that in the registration of a change of state, the event that causes the change of state is also registered and the object state (*state* attribute | active or nor active). In this modelling the attributes affected by the change of state are registered when an event causes changes in the object, where a global set of tuples with five attributes is maintained. These attributes are: the identity attribute of an object, the event that causes the change of state (value change of one of the objects attributes), the attribute affected by the change of state, and the new value of that attribute and the point in time of the event.

A object class *Object_Class* exists, with “normal attributes” and “normal methods” (non-spatial), *AttrId* (Identity attribute) *Attr₁*, *Attr₂*,..., *Attr_n*, and *Method₁*, *Method₂*,..., *Method_n*. The spatial attribute *SpatialAttr* of type *Spatial* and the Spatio-Temporal methods (events) presented in *Section 2.2* are considered as part of the *Object_Class* class too. The *TimeAttr* attribute is used as a temporal attribute and is

associated with the current time of the object. The *New*, *Delete* and *Change* methods have a direct performance only on one object (the current and/or affected object), while the *Split* and *Merge* methods work with two affected objects (one of them is the current object). The *Split* event uses an input parameter which belongs to the newly created object from the event. The *Merge* event uses an input parameter which belongs to the object that is spatially merged with the affected object. All the methods have the input variable *TimeEvent* (*TimeEvent?*) and as a precondition *TimeEvent?* > *TimeAttr*. Initially in the *Object_Class* class, *TimeAttr* is equal to an initial value *TimeVal*.

Figures 2, 3, and 4 show a formal Object-Z spatio-temporal modelling using the (state+event) approach for the *Object_Class* class. In this modelling, the five spatio-temporal events analyzed in *Section 2* are presented.

```

*****Formal Modelling Focuses Events + States*****
**1 = New, 2 = Destruction, 3 = Change, 4 = Split, 5 = Merge
Event = {1, 2, 3, 4, 5}

[Typed], [Type1], [Type2], ..., [Typen], [Spatial], [Time]
NameAttr == < attr1, attr2, ..., attrn, spatialattr >

Attributes == Typed × Type1 × Type2 × ... × Typen × Spatial
|
| ∇ X, Y : Object_Class • ( X.Attd = Y.Attd ⇒ X = Y )
|
| RegFunc : Seq ( Typed × Event × NameAttr × ( Type1 | Type2 | ... | Typen | Spatial ) × Time )
|
|-----
| Object_Class
|-----
|
| AttrId: Typed
|
| Attr1: Type1
| ...
| Attrn: Typen
|
| SpatialAttr: Spatial
| TimeAttr: Time
|
| ** Initial Values → Constants
| Vald: Typed, Val1: Type1, Val2: Type2, ..., Valn: Typen, TimeVal: Time
| SpatialVal: Spatial
|
| **State = 1 → Active Object, State = 0 → Not Active Object
| State: {0, 1}
|-----
| INIT
|-----
|
| AttrId = Vald
|
| Attr1 = Val1
| ...
| Attrn = Valn
|
| SpatialAttr = SpatialVal
|
| TimeAttr = TimeVal
| State = 1
|
| RegFunc = RegFunc ^ { Self.AttrId, 1, NameAttr 1, Vald, TimeAttr }
| RegFunc = RegFunc ^ { Self.AttrId, 1, NameAttr 2, Val1, TimeAttr }

```

Fig. 1. Object-Z (States+Events) Object_Class Modelling Approach.

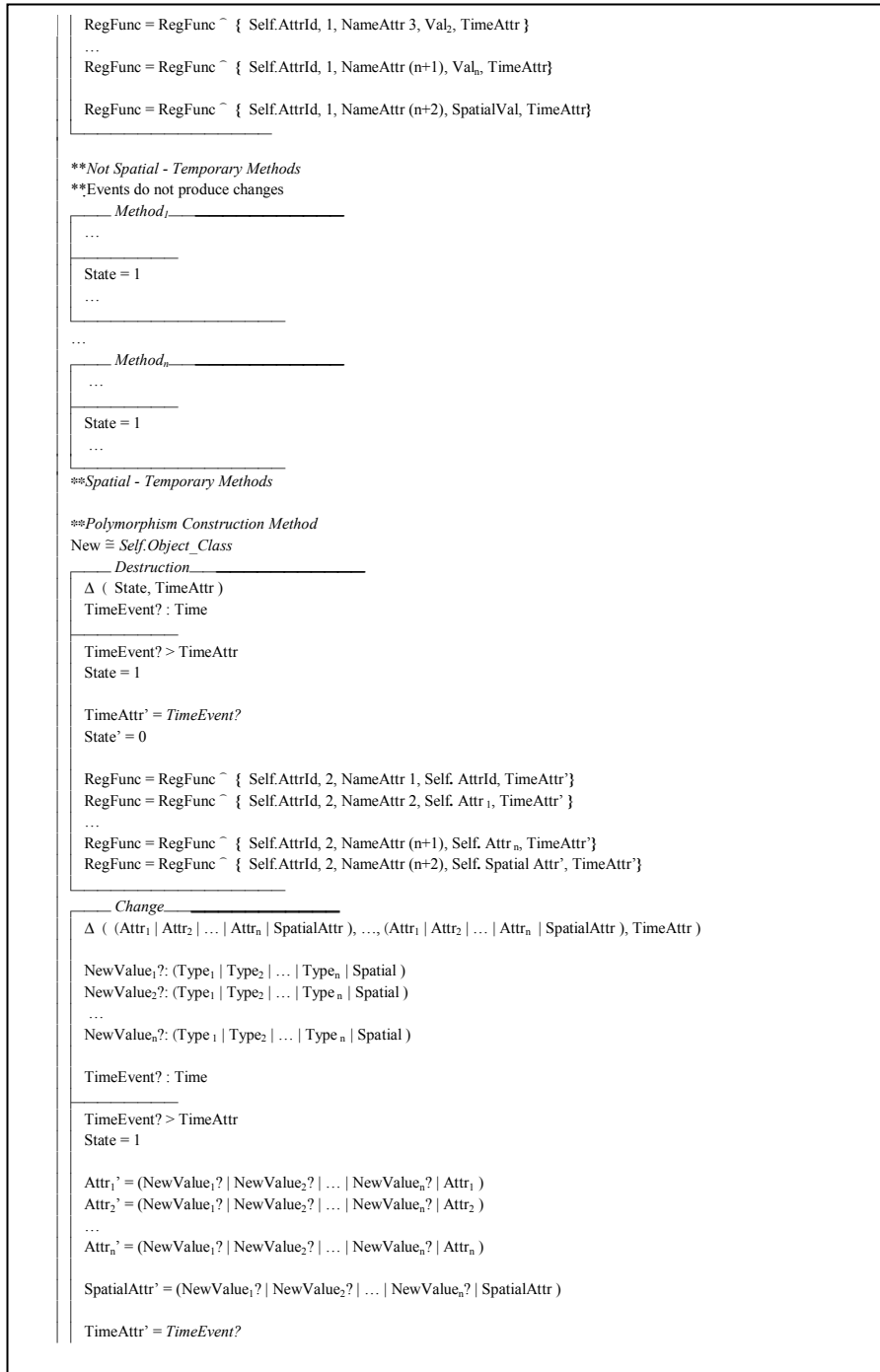


Fig. 2. Object-Z (States+Events) Object_Class Modelling Approach (Cont...).

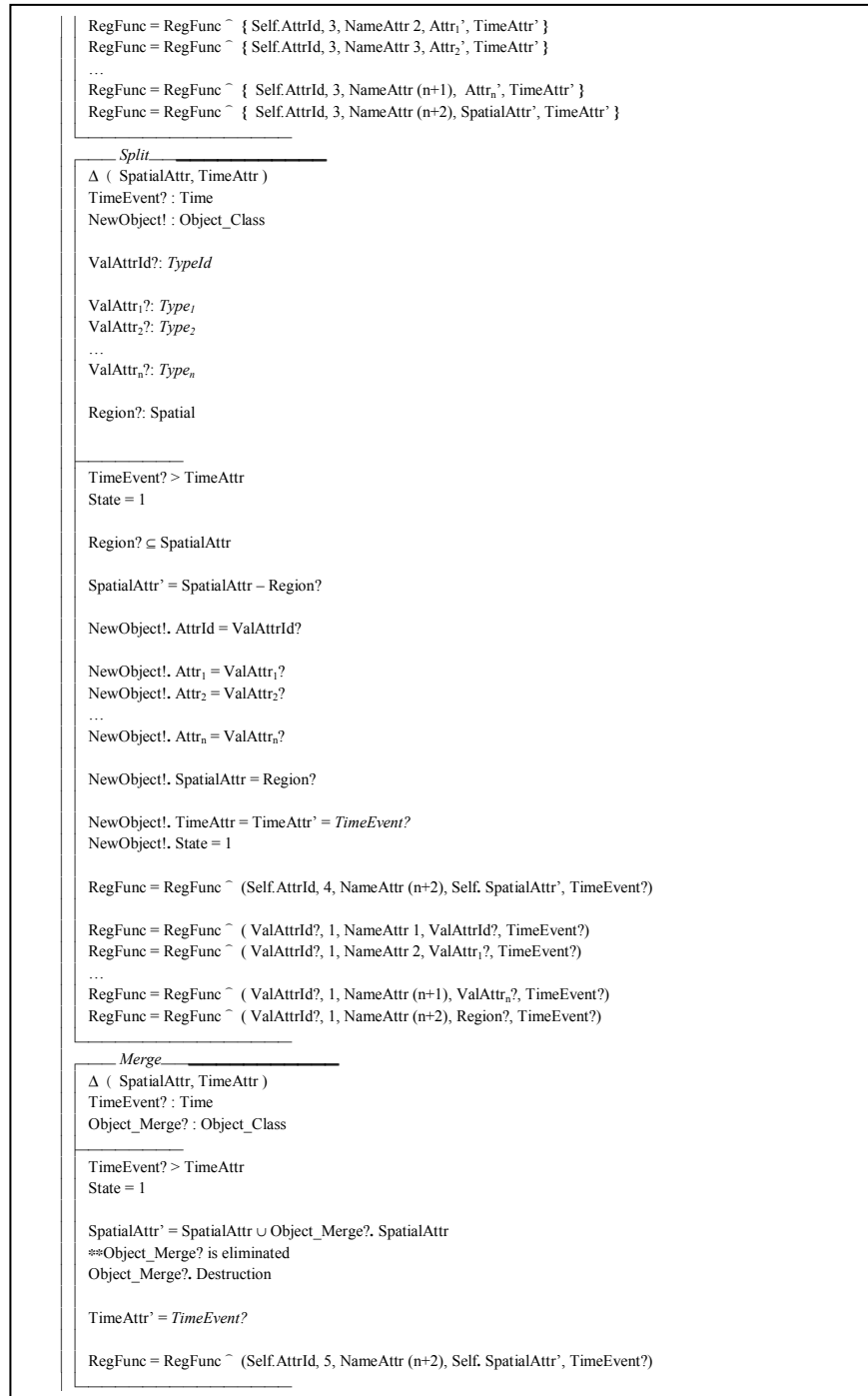


Fig. 3. Object-Z (States+Events) Object_Class Modelling Approach (Cont...).

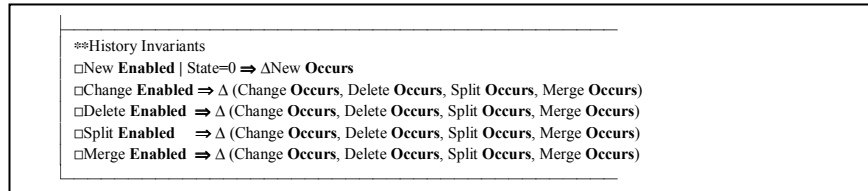


Fig. 4. Object-Z (States+Events) Object_Class Modelling Approach (Cont...).

This modelling Object-Z, it defines the class Object_Class, with methods for the spatio-temporals events described in the Section 2.2. Unvariants of history are indicated for these methods/events spatio-temporals.

With this formal modelling the semantics are detailed of each one of these events, this way not to generate mistakes, and not to produce false interpretations.

5 Conclusions

The formal modelling of information systems, is very appropriate in the development of critical systems, where a high cost is involved in there development and operation. The spatio-temporal information systems are an example of these high cost systems, because these involve directly and indirectly high cost resources, from geographic resources up to what exists or inhabits in these resources. According to studies performed in software engineering, the utilization of software development formal techniques is necessary for developing high cost or critical systems, and the most important and influential is the requirement specifications phase of a software application. Because of this, in this work the formal modelling of a spatio-temporal information system is presented, using the described approaches for its modelling and implementation. Object-Z was used as a formal specification language, taking advantage of object orientation, for a future implementation of the specified systems. The most complex inconvenient in the formal modelling of software systems, is the mathematic notation used. However this notation is not complex in general terms, since a basic knowledge of mathematical logics and set theory, is needed in a formal Object-Z specification.

The specification presented in this work, demonstrate the formal modelling by using directly and indirectly the modelling approaches described in *Section 2*. The spatio-temporal snapshot-based modelling approach is the simplest of these approaches, while the state-based and (state+event)-based approaches are more complex, due to the methods that must be specified as a part of the modelled classes. As a future work, extensions to the presented modelling are expected, to be able to determine the period of existence of an object and valid period of the state of the object, also the incorporation of other modelling frames to include other features described in some of the modelling approaches (Initial – Final time marks associated to the states of the object, other spatio-temporal queries). The complete implementation of a spatio-temporal system has been considered, using the formal modelling presented in *Section 4*, and with this the ability to improve and validate the formal specification proposed in this work.

References

- Vidal, C. (2005). A Logical Approach for Modelling Spatio-Temporal Objects and Events. *Perspectives in Conceptual Modelling, ComoGIS'05*, pages 218–227. Springer-Verlag, LNCS.
- Kesim, N. (1996). A logic programming framework for modelling temporal objects. *IEEE Transactions on Knowledge and Data Engineering*, 8(5):724–741.
- Smith, G. P. (1992). An Object-Oriented Approach to Formal Specification. PhD thesis, Dept. of Computer Science, Univeristy of Queensland.
- Davies (1999). *Using Z*. Prentice-Hall.
- Armstrong, P. (1988). Temporality in spatial databases. *Proceedings: GIS/LIS'88*, 2:880-889.
- Worboys, M. (1994). A unified model of spatial and temporal information. *Computer Journal*, 37(1):26-34
- Petry, F. E. (2003). The object event calculus and temporal geographic information systems. *Developments in Applied Artificial Intelligence*, 799-813, IEA/AIE 2003.
- Vidal, C. (2007). A Formal Frame for Database Temporary-Space Based on Objects and Events. Theses Master in Computer Science, Dept. of Computer Science, Universidad de Concepción.
- Chen, W. (1989). C-logic of complex objects. *ACM SIGACT-SIGMOD-SIGART Symp. Principles of Database Systems*, pages 369-378.
- Gregersen, H. (1999). Temporal Entity- Relationship Models—A Survey. *IEEE Transactions on Knowledge and Data Engineering*, 11(3):464–497, 1999.
- Grenin and Smith, B. (2004). Snap and span: Towards dynamic spatial ontology. *Journal of Spatial Cognition and Computation*, 4(1):69-103.
- Grenin, P. (2002). The formal ontology of spatio-temporal reality and its formalization. *AAAI*.
- Hayes (1987). *Specification Case Studies*. International Series in Computer Science, Prentice-Hall.
- Spivey, J.M. (1992). *The Z Notation: A Reference Manual*. Prentice-Hall.
- Duke, R. (1995). Object-Z: a specification language advocated for the description of standards. *Computer Standards and Interfaces*, 17:511–533.
- Meyer, B. (1998). *Object-Oriented Software Construction*. Prentice-Hall.
- Holloway, M. (1997). *Why Engineers Should Consider Formal Methods*. NASA Langley Research Center.

ALGORITMO VORAZ PARA EMPAREJAMIENTO MAXIMO DE GRAFOS

Lujan López, Luís A. Rodríguez Maysundo Eduardo Sánchez Enríquez, Heider Y.

¹ Escuela de Informática
UNIVERSIDAD NACIONAL DE TRUJILLO

lujan.ajsp@hotmail.com, edrm@hotmail.com, heider.esencia@gmail.com

Resumen

Nuestro problema es realizar un algoritmo de optimización para hallar un emparejamiento máximo entre todos los vértices de un grafo. El método más usado para resolver este problema, usa el concepto del Árbol Alternado, a través de un algoritmo formal pero laborioso (para el caso de un grafo no bipartido). Nosotros en este trabajo desarrollamos un algoritmo voraz (basado en los grados de los vértices) con un tiempo de cómputo eficiente y eficaz para grafos simples. Para ello daremos a conocer algunos conceptos previos, claves para poder entender el desarrollo del algoritmo, y la representación en ejemplos reales. La referencia principal usada en este trabajo es la tesis (Perez Martin, 2004), de donde pudimos obtener información base para este trabajo.

1. Introducción

En la actualidad muchos problemas de la vida real han sido esquematizados a través de grafos, para poder hallarles una solución a través de la optimización. Uno de estos problemas es el emparejamiento máximo de vértices en un grafo, que se presentan en muchas situaciones en las que hay que realizar asignaciones o emparejamientos sujetos a restricciones, por ejemplo, profesores y cursos a impartir, pilotos y horarios de vuelo, mujeres y posibles maridos. Como habíamos ya hablado en el resumen, el algoritmo que se va a desarrollar es de tipo Voraz, que trata de hallar la solución mas optima, a través de unos pasos generales en el estudio de un algoritmo voraz. Este algoritmo es eficaz para todo tipo de grafos simples y lo mas importante fácil de entender y poder programar en un computador; talvez desde el punto de vista matemático no parezca tan formal ya que se trata de un algoritmo voraz y por lo general estos tipos de algoritmos son heurísticos. El paper está organizado de la siguiente manera: en la sección 2 se describen los conceptos de interés para el planteamiento del problema, en la sección 3 se presentan los trabajos previos realizados, resaltando el algoritmo basado en el árbol alternador, el desarrollo de nuestra propuesta se encuentra en la sección 4 y terminamos con los experimentos y resultados obtenidos en la sección 5 y la discusión de los mismos en la sección 6.

2. Conceptos de interés

A continuación se detallarán varias definiciones sobre grafos relacionados al emparejamiento. Las siguientes definiciones son necesarias para comprender correctamente el algoritmo sobre emparejamiento. (West, 2001).

2.1. Emparejamiento

Un emparejamiento (matching) en un grafo no dirigido y simple $G = (V, E)$ es un subconjunto de aristas $M \subseteq E$ tal que dos aristas cualesquiera de M no tengan un extremo común. Evidentemente un subconjunto de un emparejamiento es también un emparejamiento aunque de menor cardinal (Figura 1).

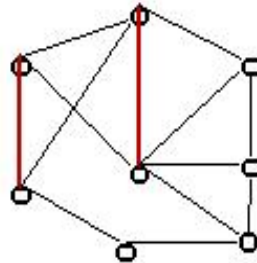


Figura 1: Emparejamiento M

2.2. Vértice saturado y vértice libre

Dado un emparejamiento M en G , los vértices incidentes con alguna arista de M se denominan vértices saturados por M o M -saturados. El resto de los vértices que no cumplen dicha condición se llaman vértices no saturados o insaturados por M , aunque el nombre más común que se les asigna es el de vértice libre.

2.3. Pareja de un vértice

Se denomina como pareja de un vértice (v_1) al vértice (v_2) del extremo opuesto en la arista perteneciente al emparejamiento M que tiene como extremo al primero de los vértices, otra forma de decirlo es que el vértice v_1 está emparejado con el vértice v_2 .

2.4. Emparejamiento perfecto

Un emparejamiento es perfecto cuando todos los vértices de G son extremo de alguna arista de M , es decir, cuando todos los vértices del grafo son vértices saturados por M .

2.5. Emparejamiento Máximal

Un emparejamiento máximal en un grafo es un emparejamiento que no puede ser ampliado agregando una arista. Un emparejamiento es máximal si y solo si toda arista de G comparte extremos con alguna arista de M . (Figura 2)

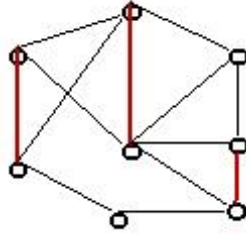


Figura 2: Emparejamiento Máximal

2.6. Emparejamiento máximo

Un emparejamiento máximo es un emparejamiento que tiene el mayor número posible de aristas. Todo emparejamiento máximo es también máximal, pero no necesariamente al revés, ya que un emparejamiento sea máximal no implica que sea máximo.

Un emparejamiento máximo puede ser perfecto (Figura 3) cuando cubre todos los vértices, de caso contrario es emparejamiento máximo no perfecto (Figura 4).

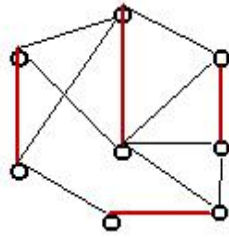


Figura 3: Emparejamiento máximo y Perfecto

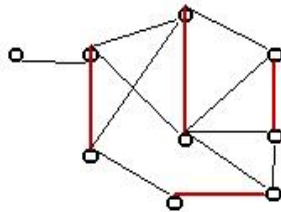


Figura 4: Emparejamiento Máximo - No Perfecto

2.7. Árbol alternado

Es una clase de árbol con raíz específico que se usa para la creación de caminos alternados utilizados para encontrar emparejamientos lo más grandes posibles de un grafo ((Hernández Peñalver, 2005))(Figura 5). Las características que tienen son:

- La raíz del árbol debe ser un vértice libre.
- Las ramas del árbol estarán formadas por aristas que, alternativamente, están o no en el emparejamiento M .

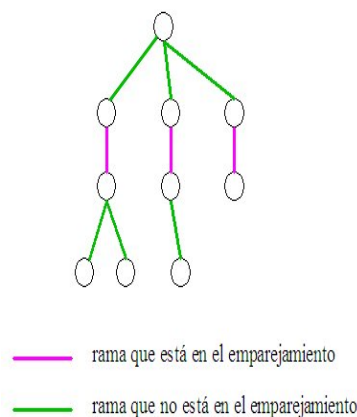


Figura 5: Ejemplo de árbol alternado

3. Trabajos previos

3.1. Algoritmos y su Complejidad

El primer algoritmo polinómico para determinar un emparejamiento máximo lo dio Edmonds en 1965. Utilizó, como todos los que habían tratado el problema en aquella época, caminos aumentadores. Edmonds dijo que su algoritmo tenía complejidad $O(n^4)$, aunque él no entró en detalles. Sin embargo Gabon y Lawer en 1976 probaron que si se implementaba el algoritmo convenientemente se reducía a $O(n^3)$.

Por otra parte, el primer algoritmo que resolvió el problema de emparejamiento máximo con una complejidad inferior a la cúbica fue el algoritmo de Even-Kariv en 1975 con una complejidad de $O(n^{5/2})$.

El mejor algoritmo en estos momentos se ejecuta en un tiempo de $O(n^{1/2}m)$ para un grafo con n vértice y m aristas, es decir, más rápido que el anterior para grafos dispersos. Este algoritmo es bastante complicado y es el algoritmo de Micali-Vazirani.

3.2. Método tradicional usando árbol alternado

El método mas usado para el emparejamiento máximo de grafos, usa el concepto del Árbol Alternado. Este método utiliza dos algoritmos, uno para grafos bipartidos y otro para grafos generales. La complejidad computacional para el primero es más eficiente y legible, por esta razón describiremos los pasos generales que considera este algoritmo, pero antes mencionaremos dos teoremas importantes que utiliza el algoritmo:

Teorema de Hall: Un grafo bipartido G con bipartición X, Y , tiene un emparejamiento que satura X si y solo si $|N(S)| \geq |S|$ para todo $S \subseteq X$. A esto se le conoce como la condición de Hall.

Teorema de Berge: Un emparejamiento M en un grafo G es máximo si y solo si G no tiene ningún M - camino de aumento.

A continuación se detallará el algoritmo encargado de buscar un emparejamiento en grafos bipartidos, pero antes se enunciarán varias observaciones interesantes que se aplican sobre el

algoritmo que se detallará seguidamente.

Si $G = (V, A)$ es un grafo bipartido con $V = X \cup Y$, y M es un emparejamiento maximal de G ; se tiene que $|M| \leq \min \{|X|, |Y|\}$.

Si $G = (V, A)$ es un grafo bipartido con $V = X \cup Y$, y M es un emparejamiento completo de G ; se tiene que $|M| = \min \{|X|, |Y|\}$.

Algoritmo 1 Emparejamiento en Grafos Bipartidos basado el el Árbol Alternador

Requiere:

Un grafo simple y bipartido G con una bipartición X, Y .

Asegurar: Un emparejamiento de cardinal máximo.

1. Tomar un emparejamiento cualesquiera M . En este paso se da la opción al usuario de elegir interactivamente el emparejamiento inicial, el cual se tomará como paso inicial para realizar el algoritmo. También está la opción de no seleccionar ningún emparejamiento inicial, en este caso se generará un emparejamiento inicial según un algoritmo que se explicará a continuación.
 2. Buscar un camino de M -aumento, mediante la construcción del árbol alternado, a continuación también se explicará el algoritmo que construye el árbol alternado y encuentra un camino de M -aumento. Si existe un camino M -aumento ir al paso 3. Si no existe ir al paso 4.
 3. Construir un emparejamiento M' con $|M'| = |M| + 1$, asignar a M el nuevo emparejamiento ampliado M' y volver al paso 2.
 4. Devolver el emparejamiento M como emparejamiento máximo.
-

La complejidad de este algoritmo es analizado por (Perez Martin, 2004) y es de orden $O(n^3)$. Para desarrollar el algoritmo para un grafo general, se añade otros conceptos mas, y el tiempo de procesamiento crece aunque la complejidad sigue siendo $O(n^3)$.

4. Desarrollo del algoritmo voraz

4.1. Presentación del algoritmo

Para desarrollar este algoritmo tomamos el concepto de estabilidad de grafos, en donde el vértice que tiene menos probabilidad de ser escogido (menor grado), es quien escoge su pareja, quien también cumple con la condición (debe ser el vértice adyacente con menor grado posible). De esa manera, emparejando desde los vértices con menor grado, se trata de formar el emparejamiento máximo en un grafo.

A continuación presentamos el algoritmo que resuelve el problema del emparejamiento máximo, teniendo en cuenta los pasos estándar de un algoritmo voraz:

1. Inicializar los grados de cada vértice, y los candidatos con todas las aristas del grafo.
2. sea v_1 el vértice de menor grado que no este saturado por M .

3. y v_2 su pareja de menor grado posible.
4. si los vértice son factibles ir al paso 5, si no rechazar dicho vértice y retornar al paso 2.
5. agregar a la solución. La arista (v_1, v_2) .
6. sacar de los candidatos todas las aristas incidentes con dichos vértices (v_1, v_2) .
7. retornar al paso 2, si aun hay aristas para valuar en los candidatos; si no lo hay, ir al paso 8.
8. retornar el conjunto solución, y fin del algoritmo.

Como se puede observar en este algoritmo, hace mención a dos conjuntos: el conjunto solución y el conjunto de candidatos; y también escoge un par de vértice y si son factibles la añade a la solución y los rechaza del conjunto de candidatos, y esto continúa hasta que el cardinal del conjunto de candidatos sea 0, y al final retorna la solución que contiene al emparejamiento máximo. Podemos comprender mas sobre este proceso con el pseudocódigo presentado en el algoritmo 2 :

4.2. Complejidad:

Si analizamos la complejidad computacional de este algoritmo, vemos que depende de dos factores primordiales: el número de Vértices y el número de aristas. Entonces su tiempo de procesamiento aproximado es el siguiente:

$$T(n) = 2n + n \log(m) + \log^2(m) \quad (1)$$

, donde n es el numero de vértices y m es el numero de aristas, por lo tanto el Orden del tiempo de complejidad se aproxima a $O(n) = (n \log(m))$, pero puede ser $O(n^2)$ si el grafo es mas denso.

4.3. Variación par el Grafo Bipartido

El algoritmo por ser general se aplica también para grafos bipartidos, pero se puede reducir la complejidad cambiando ciertos pasos:

1. inicializar los grados de cada vértice, y los candidatos con todas las aristas del grafo
2. sea v_1 el vértice de grado 1 que no este saturado por M .
3. y v_2 su única pareja.
4. si los vértice son factibles (si existe un vértice de grado 1) ir al paso 5; si no fuera así: sacar la primera arista $= (v_1, v_2)$ de los candidatos (no hay ninguna problema en hacer esto ya que los ciclos en un grafo bipartido son pares), y pasar al siguiente paso.
5. agregar a la solución. La arista (v_1, v_2) .
6. sacar de los candidatos todas las aristas incidentes con dichos vértices (v_1, v_2) .

Algoritmo 2 Emparejamiento Maximo

Requiere: Un grafo simple $G(V, A)$

Asegurar: Un emparejamiento máximo M , donde $M \subset A$

Se inicializan variables importantes como: $VG[]$ un vector de tamaño $|V|$, en donde se almacenaran los grados de cada vértice; C es el conjunto de candidatos, y M la solución (C y M almacenan aristas).

$C \leftarrow A$

$M \leftarrow \phi$

$VG_{[0..n]} = [0, \dots, 0]$

para $i \leftarrow 0$ **hasta** $i = |E|$ **hacer**

 tomar una arista de C ; $(v1, v2) \leftarrow C_{[i]}$

$VG[v1] \leftarrow VG[v1] + 1$

$VG[v2] \leftarrow VG[v2] + 1$

fin para

mientras $|C| \neq \phi$ **hacer**

$v1$ = vértice no saturado por M y de menor grado en VG

$v2$ = vértice no saturada por M y pareja de $V1$, tomada de C , con menor grado de todos los adyacentes a $V1$

si $(v1, v2)$ es factible, ósea si son incidentes a una arista no saturada por M **entonces**

$M = M + (v1, v2)$

$C = C - (v1, v2)$

$C = C - \{a = (v_i, v_j), v1 \in a \vee v2 \in a\}$

fin si

fin mientras

Retornar M , como emparejamiento máximo de G

7. retornar al paso 2, si aun hay aristas para valuar en los candidatos; si no lo hay, ir al paso 8.
8. retornar el conjunto solución, y fin del algoritmo.

5. Experimentos y Resultados

Tomemos el siguiente grafo de la figura 6 :

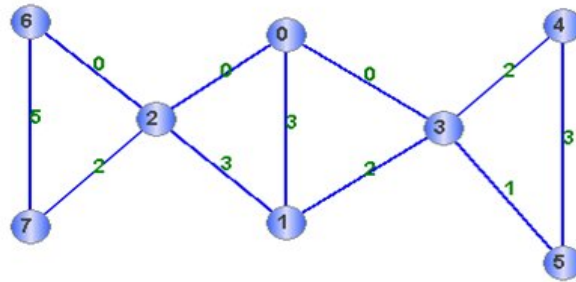


Figura 6: Grafo de aplicación

Donde: $V = \{0, 1, 2, 3, 4, 5, 6, 7\}$.

$$A = \{(0, 1), (0, 2), (0, 3), (1, 2), (1, 3), (2, 6), (2, 7), (3, 4), (3, 5), (4, 5), (6, 7)\}$$

Aplicando el algoritmo:

Paso 1:

$$C = (0, 1), (0, 2), (0, 3), (1, 2), (1, 3), (2, 6), (2, 7), (3, 4), (3, 5), (4, 5), (6, 7)$$

$$M = \{\}$$

Los grados de todos los vértices son:

$$gra(0) = 3, gra(1) = 3, gra(2) = 4, gra(3) = 4, gra(4) = 2, gra(5) = 2, gra(6) = 2, gra(7) = 2$$

Entonces el vector de grados es como sigue:

$VG = \{3, 3, 4, 4, 2, 2, 2, 2\}$, cada posición se almacenan los grados de cada vértice siguiendo el orden.

Paso 2: Ahora se escoge el vértice de menor grado, en este caso hay cuatro vértices de grado 2, entonces por orden lexicográfico escojo el vértice $V1=4$. Su menor adyacente sería el vértice $V2=5$, entonces emparejamos dichos vértices y añadimos a M . y posteriormente le quitamos de C y todas las aristas incidentes a los vértices 4 y 5.

$$M = \{(4, 5)\}$$

$$C = \{(0, 1), (0, 2), (0, 3), (1, 2), (1, 3), (2, 6), (2, 7), (3, 4), (3, 5), (4, 5), (6, 7)\}$$

Entonces $C = \{(0, 1), (0, 2), (0, 3), (1, 2), (1, 3), (2, 6), (2, 7), (6, 7)\}$

El vector de grados se actualiza: $VG = \{3, 3, 4, 2, -1, -1, 2, 2\}$ se coloca con -1 a la posición del vértice 4 y 5 para indicar que ya son saturados por M y no deben considerarse para la siguiente iteración.

Regresamos al bucle puesto que todavía hay elementos en la lista de candidatos.

Paso 3: Tomamos ahora el vértice de menor grado 6 y su menor adyacente 7. Entonces añadimos a la solución:

$$M = \{(4, 5), (6, 7)\}$$

$$C = \{(0, 1), (0, 2), (0, 3), (1, 2), (1, 3), (2, 6), (2, 7), (6, 7)\}$$

$$\text{Entonces } C = \{(0, 1), (0, 2), (0, 3), (1, 2), (1, 3)\}$$

El vector de grados se actualiza: $VG = \{3, 3, 2, 2, -1, -1, -1, -1\}$, . Regresamos al bucle puesto que todavía hay elementos en la lista de candidatos.

Paso 4: Solo nos quedan 4 vértices no saturados, escogemos el de menor grado, y es el vértice 2 y su menor adyacente sería 0, por orden lexicográfico.

$$M = \{(4, 5), (6, 7), (0, 2)\}$$

$$C = \{(0, 1), (0, 2), (0, 3), (1, 2), (1, 3)\}$$

$$\text{Entonces } C = \{(1, 3)\}$$

El vector de grados se actualiza: $VG = \{-1, 1, -1, 1, -1, -1, -1, -1\}$, Regresamos al bucle puesto que todavía hay elementos en la lista de candidatos.

Paso 5: Y solo nos queda una arista como forma un emparejamiento es tomada por M:

$$M = \{(4, 5), (6, 7), (0, 2), (1, 3)\}$$

$$C = \{\}$$

Entonces $C = \phi$. Y termina el bucle.

→ El emparejamiento M encontrado es máximo y perfecto ya que consume todos los vértices del Grafo original.

Gráficamente el emparejamiento M quedaría representado en la figura 7

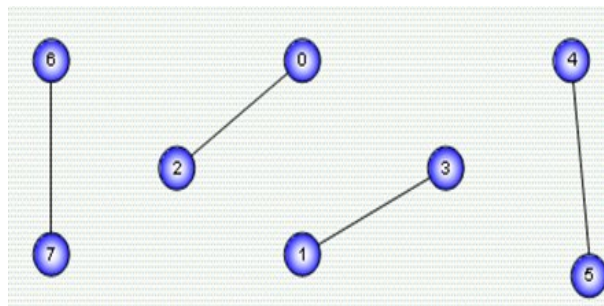


Figura 7: Emparejamiento resultante M

6. Conclusiones

- El emparejamiento de grafos es un tema de estudio nuevo pero de grandes aplicaciones, por ello necesita de un algoritmo eficiente y efectivo para la resolución de los diversos problemas sobre emparejamientos.

- El algoritmo voraz para resolver el problema del emparejamiento máximo de vértices en un grafo, es una solución adecuada por su efectividad y legibilidad.
- La complejidad computacional promedio del algoritmo voraz es de $O(n \log(n))$, pero puede llegar hasta $O(n^2)$ para grafos mas densos.
- Para el caso de grafos bipartidos el tiempo de procesamiento es mejor ya que el algoritmo para este caso modifica algunos pasos, para hacerlo mas eficiente.

Referencias

Hernández Peñalver, G. (2005). Emparejamientos y flujos. UPM.

Perez Martin, R. (2004). Emparejamiento de grafos. Trabajo de fin de Carrera - Santiago de Chile.

West, D. B. (2001). Introduction to graph theory. Prentice Hall.

Comparación de dos algoritmos metaheurísticos para el problema de optimización de portafolios de inversión

Pablo Fonseca Arroyo¹ Manuel Luna Trujillo¹ Juan Trelles Trabucco¹

¹ Pontificia Universidad Católica del Perú

pfonseca@pucp.edu.pe, a20050247@pucp.edu.pe, juan.trelles@pucp.edu.pe

Resumen

El problema de selección de proyectos de inversión para conformar portafolios óptimos de inversión es una instancia de un problema NP-completo conocido como el 0/1 knapsack problem. En el presente trabajo se comparan dos algoritmos metaheurísticos que lo solucionan. El primero es un algoritmo genético y el segundo un algoritmo tabú, ambos alimentados por un algoritmo Greedy Randomized Adaptive Search Procedure (GRASP). También se presentan las técnicas usadas para la evaluación de proyectos de inversión situados en un contexto de aversión al riesgo variable y comparación con un entorno económico.

1. Introducción

La selección de proyectos de inversión para conformar portafolios óptimos de inversión ha tomado mucha importancia en los últimos años. Esto es debido a que los inversionistas esperan una mayor eficiencia del capital cada vez. Sin embargo, las condiciones que muestra la realidad imponen distintas variables que representan los riesgos de la inversión. Estos riesgos son evaluados y finalmente contrastados con un nivel de aversión al riesgo propios a los inversionistas o los portafolios de inversión.

En el contexto local, la decisión de qué proyectos seleccionar se hace por lo general de manera manual o apoyados en hojas de cálculo. Por un lado esto es posible para una instancia pequeña de prospectos, sin embargo se torna imposible para instancias más grandes. Por esta razón, una solución automatizada y que provea un buen ratio de velocidad - coste computacional es necesaria.

El problema de selección de proyectos de inversión es una instancia del 0/1 knapsack problem. Este problema es NP-completo, por lo que se justifica el uso de técnicas aproximadas como las metaheurísticas. El presente trabajo muestra primero la definición del problema, las técnicas de evaluación usadas incluyendo una función de calificación financiera que tiene en cuenta la aversión al riesgo propia del portafolio o el inversionista, la solución mediante un algoritmo genético alimentado por un algoritmo Greedy Randomized Adaptive Search Procedure (GRASP), la solución mediante un algoritmo tabú también alimentado por el algoritmo GRASP y finalmente la experimentación numérica y conclusiones finales.

El resto del trabajo está dividido como sigue: en la sección 2 se muestran los trabajos previos que han intentado resolver el problema. Luego, en la sección 3 se define de manera más específica el problema. En la sección 4 se muestra la evaluación de proyectos y sus particularidades. Posteriormente, en la sección 5 se muestra el algoritmo GRASP que alimenta ambos algoritmos. Ya en la sección 6 se muestra el algoritmo genético, para luego pasar en la sección 7 al desarrollo del algoritmo tabú. En la sección 8 se muestran los resultados de la experimentación numérica y en la sección 9 se discuten estos resultados. Y para finalizar se muestran las conclusiones de la investigación en la sección 10.

2. Trabajos Previos

El problema de la elección del portafolio de inversión ideal ha sido tratado múltiples veces por diversos investigadores y existen muchas otras soluciones comerciales que aseguran resolverlo eficientemente. Las aproximaciones hechas para resolver este problema han usado una amplia variedad de algoritmos, siendo los preferidos los algoritmos genéticos.

La mayoría de los enfoques están centrados en la resolución del problema de Markowitz en un contexto de mercado de valores, cuyas variables resultan diversas al contexto de proyectos de inversión en construcción pero cuyas soluciones ayudaron a ampliar el conocimiento sobre estos tipos de algoritmos.

La solución de Kin Keung Lai, Lean Yu, Shouyang Wang y Chengxiong Zhou (Lai et al., 2006), utiliza dos algoritmos genéticos para generar un ranking de acciones y luego realizar la optimización. Esta solución, como muchas otras, se centró en la covarianza y utilizó un cromosoma binario. Chi Min Ling (Lin, 2007) ofrece un análisis multiobjetivo, en el que primero evalúa el desempeño histórico de un grupo de acciones para luego ponderarlas; es interesante mencionar que se utilizan cromosomas formados por genes de números decimales. Se puede también encontrar soluciones complejas como el algoritmo genético de Wei-Guo Zhang, Wei Chen y Ying-Luo Wang (Wei-Guo Zhang, 2007) basados en modelos estocásticos.

3. Definición del problema

El problema de selección de proyectos de inversión es finalmente una instancia del problema de complejidad exponencial conocido como el 0/1 knapsack problem. Este problema es un caso especial de programación entera (Sahni, 1975). Se definen dos conjuntos. $P = \{p_1, p_2, p_3 \dots p_r\}$ y $S = \{s_1, s_2, s_3 \dots s_r\}$, donde p_i es la ganancia asociada al objeto i -ésimo y s_i es el tamaño o peso del objeto i -ésimo. La mochila tiene una capacidad C y se seleccionan los objetos que maximicen la ganancia asociada sujetos a la restricción de capacidad.

4. Evaluación de proyectos

Para evaluar los riesgos de un proyecto se debe tener una cantidad de variables que muestren las distintas características del proyecto. Un evaluador humano observaría estas características y en base a la experiencia y la técnica, asignaría un puntaje de calificación financiera. Para el presente trabajo se ha considerado la utilización de un árbol de decisiones, que asigna una ponderación jerárquica a las características del proyecto. De este modo se puede modelar la aversión al riesgo de manera más detallada y real. Cada hoja del árbol asumirá un valor de puntuación entre 1 y M , donde $M > 1$ y además M representa un valor más deseable. Cada rama, a su vez, tendrá un valor entre 0 y 1. Los valores de las ramas de un mismo nivel deben sumar 1 siempre. A continuación la Figura 1 muestra la construcción de un árbol para el caso particular de proyectos de construcción inmobiliarios.

Para evaluar las distintas características se siguieron algunas pautas. Si la evaluación está en función al entorno económico que se analiza, se opta por asignar una puntuación de acuerdo al medio. En los casos en que lo ideal sea encontrarse cerca a la media del sector, se usa la distribución normal para computar la calificación. Se procede a estandarizar la característica tratada de acuerdo a la media y desviación estándar del sector. En la Figura 2 se muestra la gráfica de la función de probabilidad de la distribución normal. Puede notarse que los valores

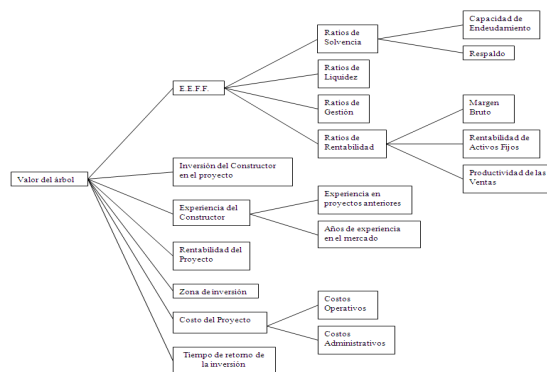


Figura 1: Árbol de decisiones para proyectos de construcción

cercanos a la media tendrán un valor mayor. Por el teorema del límite central se puede asumir que la distribución de probabilidades del sector analizado seguirá una distribución normal.

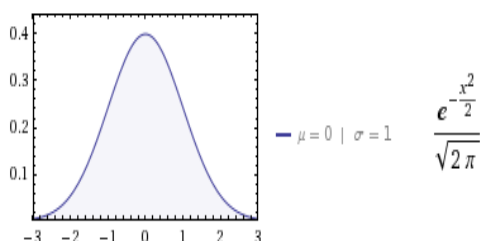


Figura 2: Gráfica de la función de Probabilidades de la distribución normal estándar

Si deseamos que el valor de la calificación varíe entre 0 y 1, obtenemos finalmente la expresión (1) a partir de la función de probabilidad de la distribución normal estándar.

$$f(x) = \frac{\exp(-x^2/2)}{\sqrt{2 * \pi}} \quad (1)$$

Si se quiere que los valores varíen entre 1 y M aplicamos una transformación lineal y obtenemos (2).

$$f(x) = 1 + \exp(-x^2/2) * (M - 1) \quad (2)$$

También hay características que deben compararse con el entorno particular que se ha configurado al tener una lista inicial de prospectos. En este caso podemos distinguir características deseadas y no deseadas. Una característica deseada podrá ser puntuada distribuyendo los valores linealmente entre 1 y M según (3).

$$f(x) = 1 + [(Valor - Menor)/(Mayor - Menor)] * (M - 1) \quad (3)$$

Mayor es el valor más alto de la característica evaluada en todos los elementos de la lista. Y de igual manera, Menor es el valor más bajo para esa característica en toda la lista.

En caso la característica sea no deseada, se distribuye linealmente de manera invertida según (4).

$$f(x) = M - [(Valor - Menor)/(Mayor - Menor)] * (M - 1) \quad (4)$$

Existe la posibilidad que hayan ciertos valores para determinadas características que hagan indeseable un proyecto. Ejemplo de esto es el VPN (Valor Presente Neto) donde un valor menor a 0 indica un proyecto que no resulta rentable en la medida que su eficiencia no supera la TMAR (Tasa mínima atractiva de retorno). En este tipo de casos, el proyecto no debe ser evaluado para no sesgar la puntuación de las características comparadas con el conjunto de prospectos. Por lo que antes de ejecutar la evaluación se filtran estos proyectos.

En la figura 3 se muestran las fases que se seguirán para la obtención final del resultado. Se puede observar un conjunto inicial de proyectos, posteriormente son filtrados. Solo ahí se optimiza el portafolio, sujeto a las restricciones de presupuesto, obteniéndose el resultado final.

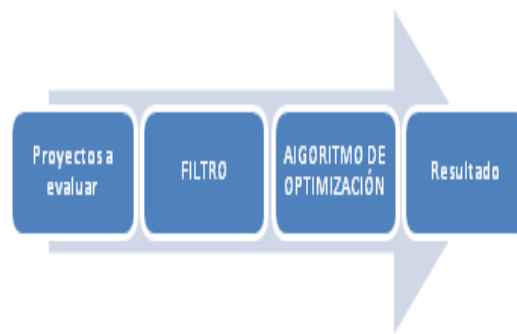


Figura 3: Esquema de la aplicación del algoritmo

La función de evaluación de proyectos asignará un puntaje a cada proyecto. Este puntaje se calcula según (5) donde P y Q son factores de ponderación para la rentabilidad del proyecto y valor resultante del árbol, respectivamente.

$$f(x) = \frac{P * rentabilidadProyecto}{Q * (M - ValorResultanteArbol)} \quad (5)$$

El valor resultante del árbol se encuentra sumando los valores multiplicados desde las hojas hasta la raíz. El puntaje finalmente representa una calificación financiera que indica la deseabilidad del proyecto acorde a la aversión al riesgo del portafolio.

5. Algoritmo GRASP - Construcción

Un algoritmo GRASP es un método constructivo. Desarrollado originalmente por Feo y Resende (Feo and Resende, 1989). Se basa en un parámetro α que se conoce como constante de relajación. La utilización de este parámetro sirve para reducir la miopía presente en los algoritmos voraces. Cada iteración en la fase de construcción del algoritmo representa la selección de un proyecto. La conclusión del algoritmo nos indica que se ha generado una lista con los proyectos seleccionados, sujetos a una restricción de presupuesto. A continuación se muestra el algoritmo GRASP-construcción usado. Cabe resaltar que se tienen que ejecutar N veces el

algoritmo GRASP-construcción para obtener un resultado que supere la instancia voraz.

Algoritmo 1 Algoritmo Grasp

mientras *presupuesto* > 0 **hacer**
Ordenar(ListaProyectos, puntaje, descendente)
 $\beta \leftarrow \text{Primero}(\text{ListaProyectos})$
 $\tau \leftarrow \text{Ultimo}(\text{ListaProyectos})$
 $RCL \leftarrow \text{FormarRCL}(\text{ListaProyectos}, \beta, \tau, \alpha)$
proyectoSeleccionado $\leftarrow \text{Aleatorio}(RCL)$
ListaProyectos $\leftarrow \text{ListaProyectos} - \{\text{proyectoSeleccionado}\}$
ListaSeleccionados $\leftarrow \text{ListaSeleccionados} \cup \{\text{proyectoSeleccionado}\}$
ReducirPresupuesto(*proyectoSeleccionado*)
fin mientras

6. Algoritmo Genético

Los algoritmos genéticos se basan en una simulación de la teoría de la evolución aplicada a diversos problemas de inteligencia artificial. Fue propuesto por Holland en los años setenta (Holland, 1975). Se intenta simular poblaciones que van evolucionando tras operaciones naturales como el casamiento y la mutación. El algoritmo genético propuesto adapta estas operaciones al contexto de optimización de portafolios de inversión.

Gen - Representa un proyecto, tiene dos posibles valores 0 y 1. Un valor de 1 indicará que el proyecto forma parte de un portafolio de inversión.

Cromosoma - Estructura que contiene todos los proyectos evaluados por el algoritmo (tras la fase de filtrado).

Cabe resaltar que la representación de un cromosoma es un arreglo de bits y que cada individuo de la población constituye un portafolio, es decir, una solución factible al problema. Se considera una aberración al individuo que tiene proyectos seleccionados por un presupuesto mayor a lo disponible para el portafolio.

En la figura 4 se muestra el esquema antes mencionado.

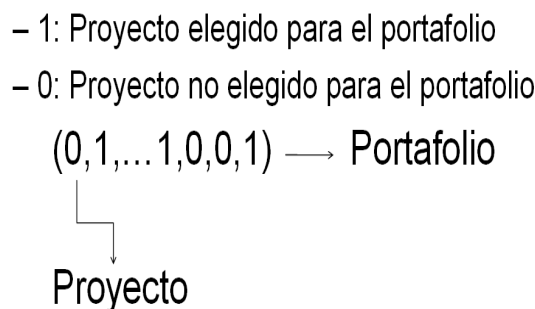


Figura 4: Esquema de las estructuras para el algoritmo genético

Operadores del algoritmo:

Casamiento - Operador que genera individuos (portafolios) en base a la combinación de las

características de algunos individuos denominados padres. Se utilizaron 2 portafolios padres para la generación de 2 portafolios hijos. La regla es que los individuos resultantes deben tener ciertas características que aseguren una correcta evolución de las poblaciones y una futura convergencia del algoritmo. Primero se tiene que el portafolio hijo no debe ser igual a alguno de los padres (tener seleccionados los mismos proyectos). Y en segundo lugar, los portafolios generados deben ser mejores que los padres. La implementación elegida de este operador consiste en partir el vector que representa a ambos padres y combinar una parte de uno con la del otro, y obtener los hijos de las dos posibles combinaciones. En la figura 5 se muestra un ejemplo de la aplicación de este operador.

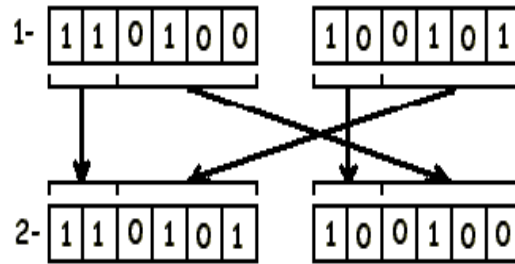


Figura 5: Operador de casamiento

Mutación - Operador que altera a unos pocos individuos de la población según una probabilidad de mutación $m\%$. Con esto se logra diversificar la evolución sin afectar la convergencia. Se visitan otros espacios de solución que no se pueden obtener a través del casamiento. Se obtiene aleatoriamente una gen del cromosoma y se invierte su valor (Un proyecto seleccionado pasa a no estarlo, o viceversa). En la figura 6 se muestra la aplicación de este operador.

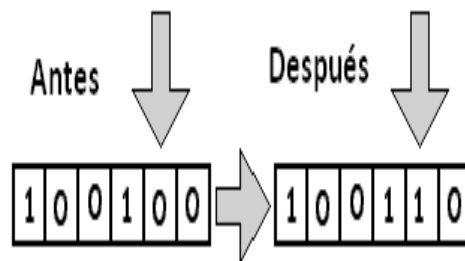


Figura 6: Operador de casamiento

Fitness o Adaptación - Para medir la adaptación de un individuo, es decir qué tan deseable es el portafolio se debe tener en consideración la deseabilidad de los proyectos. Para esto se suman los valores de calificación de los proyectos seleccionados en el portafolio, calculándolos como se ha visto en la sección 4.

Se muestra finalmente el algoritmo utilizado.

Algoritmo 2 Algoritmo Genético

$poblacion \leftarrow GeneracionPoblacionInicialDeAlgoritmoGRASP()$

mientras *condiciondeparada* **hacer**

$MedirAdaptacion(poblacion)$

$poblacionMejoresAdaptados \leftarrow SeleccionMejoresAdaptados(poblacion)$

$poblacionMejora \leftarrow Casamiento(poblacionMejoresAdaptados)$

$poblacionMutacion \leftarrow Mutacion(poblacionMejoresAdaptados)$

$poblacion \leftarrow poblacionMejoresAdaptados \cup poblacionMutacion \cup poblacionMejora$

fin mientras

7. Algoritmo Tabú

La búsqueda tabú (Tabu Search o TS) es una metaheurística planteada inicialmente por Fred Glover (Glover, 1989). Desde su aparición, numerosos artículos y estudios se han realizado para intentar resolver problemas combinatorios. La principal idea del TS es realizar búsquedas en vecindarios. Se apoya en conceptos conocidos como memoria, lista tabú, diversificación e intensificación para explorar gran parte del espacio muestral. Otra idea importante en el desarrollo de la búsqueda por vecindarios es que muchas veces ir por un camino no tan óptimo, puede ayudar a que más adelante se encuentre un vecindario más favorable; en resumen, se combate la miopía. A continuación, se describirá el algoritmo de búsqueda tabú para el problema de selección del portafolio óptimo de inversión.

Elementos del espacio muestral - El espacio muestral está formado por portafolios de inversión. Cada elemento (un portafolio) tiene la siguiente estructura: $Elemento = (x_1, x_2, x_3, \dots, x_n)$ donde cada x_i es un proyecto de inversión representado con un 1 o 0; uno indicando que se invierte en el proyecto y cero lo contrario. Un vecindario $N(x)$ se define como el portafolio al que puedo llegar al invertir o dejando de invertir en un proyecto, es decir, cambiando un uno por un cero o un cero por un uno. Existen entonces n vecinos para cualquier portafolio.

Lista tabú - La lista tabú almacena los portafolios prohibidos durante un proceso de búsqueda, es decir, portafolios que no pueden ser visitados durante un número determinado de iteraciones, valor representado por el *tabu tenure*. Esta penalización se da cuando un portafolio es visitado en un turno de búsqueda. Se intuye entonces que el objetivo de la lista es evitar evaluar repetidamente las soluciones del espacio. La lista representa a la memoria a corto plazo del TS para la solución planteada.

Memoria a largo plazo y estrategias - Estructura que almacenará los mejores K portafolios encontrados en el desarrollo de la búsqueda. Si se van a realizar F iteraciones para el TS, se plantea que el primer tercio de las mismas no se vea afectado por ninguna estrategia de diversificación o intensificación. Luego cada G iteraciones, se aplicará una estrategia consistente en comparar el valor un portafolio aleatorio de la memoria a largo plazo con el promedio del valor de las soluciones de la misma. Si el valor es menor al promedio, se procederá a limpiar la lista tabú y construir un portafolio aleatorio con el fin de explorar otra zona del universo. Caso contrario, se escoge dicho portafolio y se procede a seguir la búsqueda en ese vecindario.

Criterio de aspiración - El uso de la lista tabú ayuda a evitar caer en un óptimo local pero a veces se puede encontrar un dilema. ¿Qué sucede si todos los vecinos de un portafolio están penalizados? La respuesta es simple, se utiliza el criterio de aspiración que sugiere elegir al miembro de la lista menos tabú, es decir, aquel portafolio cuya penalización sea menor. Existe también otro caso donde se puede romper la regla de la penalización. Este se da cuando un vecino está etiquetado como tabú pero brinda la mejor solución encontrada hasta el momento; si se presenta este caso, se elige a ese portafolio y luego se vuelve a etiquetar como tabú.

Algoritmo 3 Algoritmo Búsqueda Tabú

```

solucionActual ← ObtenerSolucionDeAlgoritmoGrasp()
mientras condiciondeparada hacer
    vecindario ← GenerarVecindario(solucionActual)
    solucionActual ← MejorSolucion(vecindario)
    ActualizarHistorial()
    ActualizarTurnoTabu()
    si solucionActual > mejorSolucion entonces
        mejorSolucion ← solucionActual
    fin si
    si TurnoEstrategia() entonces
        RealizarEstrategia()
    fin si
fin mientras

```

8. Experimentos y Resultados

Se realizaron experimentos con el propósito de contrastar el desempeño de ambos algoritmos para una instancia de prueba aleatoria. En la figura 7 se muestra el esquema de experimentación. Se generaron aleatoriamente 30 instancias de selección de proyectos con 20 proyectos aleatorios. Se simularon las características de proyectos de construcción inmobiliarios (cerca de 25). Estas características dependiendo su naturaleza fueron simuladas con una distribución uniforme o una distribución normal.

Para el experimento se generaron 20 portafolios (los mejores generados por el algoritmo GRASP) y estos formaron parte de la población inicial del algoritmo genético, y el mejor de estos 20 fue la solución inicial del algoritmo tabú.

Se realizó una prueba de Lévene para verificar si las varianzas de los resultados de los algoritmos son iguales. Siendo el estadístico de contraste $F = S_1^2/S_2^2$ con $\alpha = 0,05$. El $P - valor = 0,038 < 0,05$ por lo que se puede decir que las varianzas son distintas.

Se realizó también una prueba de hipótesis acerca de dos medias y varianzas desconocidas pero distintas con un $\alpha = 0,05$. El resultado fue $T = -5,346 < -1,67$ (por lo que se concluye que el algoritmo genético tiene mejores resultados que el algoritmo de búsqueda Tabú).

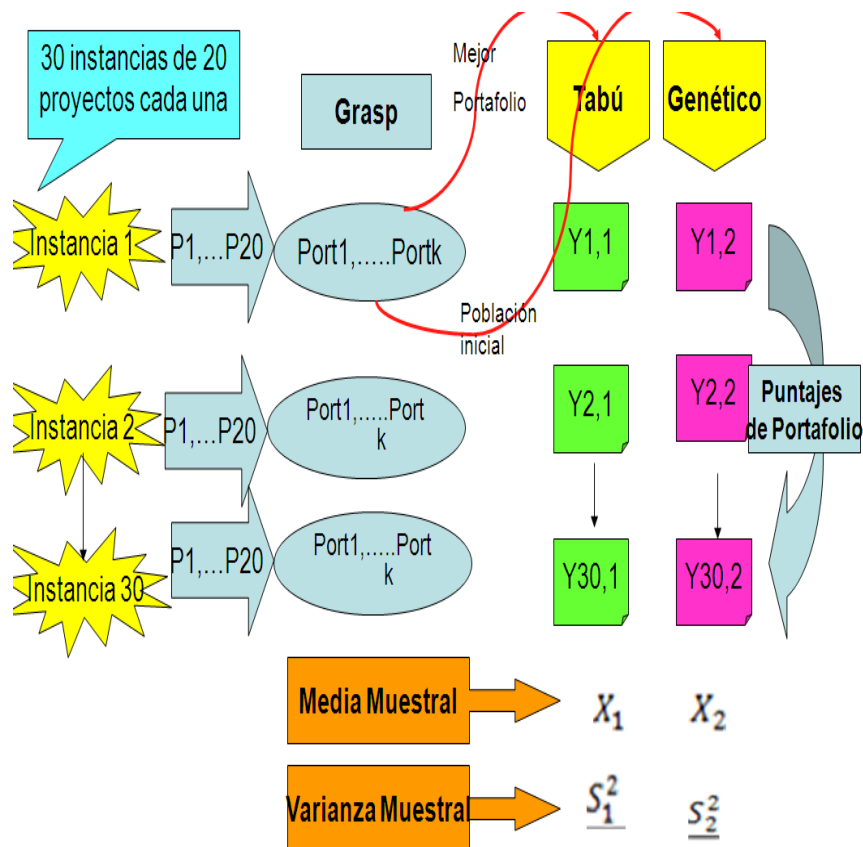


Figura 7: Esquema de la experimentación

9. Discusión de los Experimentos

El experimento mostró tras la comparación de medias que los resultados del algoritmo genético son en cuanto a comportamiento promedio, mejores que los resultados del algoritmo tabú. Sin embargo, en cuanto a tiempo de ejecución el algoritmo tabú es ligeramente más rápido.

10. Conclusiones

Tras los resultados de la experimentación se desarrolló el sistema PortMizer (Portfolio Optimizer) para proyectos de inversión en el sector de construcción inmobiliario. Este sistema es capaz de manejar instancias grandes de proyectos y muchas características con costes computacionales aceptables (Una instancia de 100 proyectos fue resuelta en menos de 1 minuto). Este esquema de trabajo se adapta a varios tipos de portafolios de inversión. En la realidad nacional es una importante alternativa para la evaluación de proyectos de inversión.

Referencias

- Feo, T. and Resende, M. (1989). A probabilistic heuristic for a computationally difficult set covering problem. *Operations Research Letters*, 8:67–71.
- Glover, F. (1989). Tabu search - part i. *INFORMS Journal on Computing*, 1(3):190–206.

- Holland (1975). *Adaptation in Natural and Artificial Systems*.
- Lai, K. K., Yu, L., Wang, S., and Zhou, C. (2006). A double-stage genetic optimization algorithm for portfolio selection. In *ICONIP (3)*, pages 928–937.
- Lin, C.-M. (2007). An effective decision-based genetic algorithm approach to multiobjective portfolio optimization problem. *Applied Mathematical Sciences*, 1(5):201–210.
- Sahni, S. (1975). Approximate algorithms for the 0/1 knapsack problem. *J. ACM*, 22(1):115–124.
- Wei-Guo Zhang, Wei CHEN, Y.-L. W. (2007). An effective decision-based genetic algorithm approach to multiobjective portfolio optimization problem. *International Journal of Computer Science and Network Security*, 6:196–200.

Evaluación Colaborativa en Grupos de Trabajo y Aprendizaje usando el Framework IssueNet

Tatiana Escovedo¹

Rubens Nascimento Melo¹

¹ Pontificia Universidade Católica do Rio de Janeiro

tatiana@inf.puc-rio.br, rubens@inf.puc-rio.br

Resumen

En la actualidad, el mercado de trabajo se caracteriza por la globalización, la fuerte competencia, los rápidos cambios, el creciente flujo y obsolescencia de la información y exigentes normas de calidad y productividad. Para acompañar estas transformaciones, la enseñanza también tiene que evolucionar desde el modelo clásico hasta el aprendizaje colaborativo, con el fin de formar individuos capaces de comunicarse, trabajar juntos para resolver problemas complejos e interdisciplinarios, coordinar el trabajo individual y del grupo, y tomar las mejores decisiones. Esta investigación trata específicamente la evaluación del trabajo y del aprendizaje colaborativo, y propone IssueNet, un Framework para la evaluación colaborativa de tareas. Para validar la contribución del Framework en la evaluación colaborativa, e investigar qué otras influencias pueda tener en grupos de trabajo o de aprendizaje se llevaron a cabo dos estudios de casos con dos instancias distintas de IssueNet. Tras el análisis de estudios de caso y testimonios de los participantes, se concluyó que el Framework satisface nuestras expectativas de permitir la evaluación colaborativa del trabajo o del aprendizaje.

1. Introducción

La Revolución Industrial marcó el inicio del proceso de rápida acumulación de bienes de capital. Iniciada en el siglo XVIII en Inglaterra, trajo el método de trabajo basado en la fragmentación de las tareas (Taylor, 1990), en el mando y control directos (Fayol, 1994) y en la producción en masa (Ford, 1993).

Actualmente, el mercado se caracteriza por la globalización, la fuerte competencia, los rápidos cambios, el creciente flujo y obsolescencia de la información y exigentes normas de calidad y productividad aumentando el flujo y la obsolescencia de la información y exigentes normas de calidad y productividad. Así, una parte creciente del trabajo de las empresas e instituciones ya no se realiza de forma individual, con una sola persona trabajando para completar las tareas. El trabajo es realizado cada vez más de forma colaborativa. Esta tendencia se debe en parte a la creciente complejidad de las tareas, que ahora requieren habilidades multidisciplinarias, y nuevos paradigmas de trabajo, que envuelven varios sectores de negocios, o incluso otras empresas, que trabajan conjuntamente en diversas etapas de preparación de un producto o el desarrollo de un proyecto (Fuks et al., 2003).

Para acompañar estas transformaciones, la enseñanza también tiene que evolucionar desde el modelo clásico para el aprendizaje colaborativo. Este desarrollo tiene como objetivo formar personas capaces de comunicarse, trabajar juntos para resolver problemas complejos e interdisciplinarios, coordinar el trabajo individual y de grupo y tomar las mejores decisiones.

Este estudio investiga específicamente la evaluación colaborativa en grupos de trabajo y aprendizaje, y propone IssueNet, un Framework de colaboración para la evaluación colaborativa de tareas. Para validar la contribución del Framework en la evaluación colaborativa, y qué otras influencias pueda ejercer en grupos de trabajo o de aprendizaje se llevaron a cabo dos estudios de caso en Brasil. Una instancia se utilizó en una clase del curso de posgrado "Proyecto de Sistemas de Software", organizada por el Departamento de Informática de la Pontificia Universidad Católica de Río de Janeiro (PUC-Río) y otro caso se ha utilizado en el grupo de usuarios de Java de Río de Janeiro (RioJUG).

2. Trabajos Previos

La investigación presentada en (Escovedo et al., 2006) ilustra cómo los alumnos aprecian y extrañan la evaluación colaborativa cuando no es utilizada. En las palabras del aprendiz que sugiere la propuesta de evaluar los mensajes del foro para un curso *online*: *"La Evaluación es una actividad muy interesante. Nos posibilita reflexionar, analizar, estudiar ... muchas, muchas otras buenas razones podrían ser alegadas para justificar esta actividad: para evaluar al otro como a sí mismo. No entiendo por qué esta actividad tan deliciosamente cognitiva - evaluar - se limita a un aperitivo para los profesores. (...) Vea lo sencillo que sería si en el ambiente estuviera disponible al final de cada mensaje de un colega, un campo para marcar la calificación que le daría usted y añadir su revisión"*.

En esta línea, (Prata, 2002) señala que para los profesores, la evaluación de los alumnos ha sido siempre un desafío, porque es difícil entender el nivel real de aprendizaje de la conjetura de los alumnos. Además, establece que la evaluación sigue siendo motivo de mucha ansiedad en los docentes - porque hay dudas sobre la calidad, lealtad, imparcialidad, integridad, honradez, justicia, privacidad, eficiencia y honestidad de la evaluación - como en los alumnos - por miedo, inseguridad, la humillación, la desconfianza, la ira, y el tumulto de la evaluación.

Así, cuando los alumnos comparten la responsabilidad de la evaluación, hay un aumento en la comprensión y la utilidad de la evaluación (O'Sullivan, 2004) (McConnell, 2002). Para los evaluados, varias evaluaciones de su trabajo generan una mayor confianza en los resultados de la evaluación, y los evaluadores comprenden mejor el proceso y los objetivos de la evaluación, aumentan su pensamiento crítico y la capacidad de mejorar su propio trabajo.

3. El Framework IssueNet

El Framework fue implementado utilizando el MVC (Model-View-Controller). El MVC es un patrón de arquitectura utilizado en la ingeniería de software, cuyo objetivo es separar los datos (Model) de la interfaz de usuario (View), de modo que los cambios en una no afecten la otra. Este patrón desacopla el acceso a datos y lógica empresarial de la presentación de datos y la interacción con el usuario, a través de la introducción del componente intermedio Controller (Gamma, 1995).

El IssueNet fue implementado sobre la base de una worklist de tareas, y la evaluación colaborativa ocurre cuando varios evaluadores evalúen una misma resolución de tarea. IssueNet presenta dos módulos principales: el módulo de Gerencia de usuarios y el módulo de Gerencia de tareas. El módulo de Gerencia de usuarios centra las operaciones relativas a los usuarios del sistema. En el Framework, se ofrecen implementaciones genéricas de usuario y perfil, que puede ampliarse según las necesidades de las instancias. Por ejemplo, pueden ser creados perfiles de acceso para delimitar las responsabilidades de cada papel de usuario, tales como profesores y alumnos. Ya el módulo Gerencia de tareas centra las operaciones relativas a las tareas del sistema. Al igual que en el módulo de Gerencia de usuarios, se ofrecen las implementaciones genéricas de tareas y de grupo de trabajo, que pueden ser extendidas. Diferentes tipos de tareas se pueden crear, por ejemplo, para posibilitar tareas de presentación de los archivos y tareas de rellenar cuestionarios.

En el Framework ya se implementa un ciclo de vida de tareas definido, que incluye los estados 'Abierta', 'Resuelta', 'En evaluación', 'Cerrada' y 'Vencida'. Sin embargo, las instancias pueden modificarlo según sea necesario, añadir y borrar estados, o incluso cambiar el orden del ciclo de vida de los estados ya implementados. Otro punto de extensión de IssueNet es la forma en que se manejan las notas recibidas en las evaluaciones. Si en una instancia es necesario calcular las notas parciales o medias, tendrá que decidir cuál es su estrategia, y aplicarla de la manera que mejor se adapte a cada caso. El Framework también proporciona mecanismos de notificación e informes, que pueden ser extendidos.

En esta sección se trató de presentar el Framework IssueNet y algunos de sus puntos flexibles. En la próxima sección se presentan en detalle los estudios de caso con dos instancias de IssueNet.

4. Experimentos y Resultados

La sección anterior presentó un Framework diseñado para apoyar la evaluación colaborativa. En esta sección se presentarán dos estudios de caso realizados para validar la contribución del Framework en este proceso.

4.1 Estudio de Caso 1: Evaluación de Trabajos Académicos

El curso "Proyecto de Sistemas de Software" (PSS) es ofrecido por el Departamento de Informática de la Pontificia Universidad Católica de Río de Janeiro (PUC-Río), en Brasil, como materia optativa en el área de Ingeniería de Software, y se ofrece para alumnos de graduación y posgrado. Su objetivo es presentar las técnicas de proyecto utilizadas para la elaboración de un software orientado al objeto, ilustrando las ventajas y desventajas de cada uno con ejemplos.

A cada semana, cada estudiante debe presentar a la clase su progreso en la preparación de un trabajo. Los otros alumnos, deben asistir a la presentación, y los maestros, evaluarlos. Al final del semestre cuando cada alumno presenta el trabajo completo, las notas de las presentaciones semanales son usadas en el cálculo de la calificación final del alumno. Esta estrategia de presentaciones semanales es adecuada para evitar que los alumnos terminen dejando para hacer todo el trabajo a última hora, ya que requiere que cada estudiante presente el progreso del trabajo a cada semana. Sin embargo, la evaluación de las presentaciones se hace únicamente por el profesor.

Como todos los alumnos asisten a todas las presentaciones, ellos también podrían contribuir con el proceso de evaluación de una presentación. La responsabilidad de la evaluación dejaría de ser centralizada en el profesor y los alumnos podrían beneficiarse de la evaluación colaborativa: el estudiante evaluado recibiría calificaciones de diferentes personas que tienen, por lo tanto, opiniones, críticas y sugerencias diferentes. Para los alumnos evaluadores, al recibir la responsabilidad de evaluar el trabajo de su compañero, estarán más atentos cuando preparen su propio trabajo, evitando repetir los errores encontrados en el trabajo de sus compañeros. Por lo tanto, hay indicios de que la evaluación colaborativa de las presentaciones de este trabajo es una buena estrategia para la disciplina de la PSS.

Se identificaron dos perfiles de acceso diferentes para este estudio de caso: profesores y alumnos. La Figura 1 ilustra la pantalla de bienvenida del perfil de acceso 'profesor', que contiene un menú al lado y los iconos de acceso directo a los principales recursos del sistema. El perfil de los alumnos es responsable por atender y evaluar las tareas, mientras que el perfil de acceso de los profesores es responsable por la gestión de usuarios, perfiles, actividades y modelos de criterios, por la creación y evaluación de tareas.



Figura 1: Pantalla de bienvenida en portugués

Para cada presentación semanal, el profesor crea una tarea en el sistema para cada alumno. En el momento de la presentación, el alumno presentador resuelve esta tarea y esta - estará disponible para evaluación. Para cada grupo de tareas semanales, el profesor definió un conjunto de criterios de evaluación que deberían ser satisfechos por los alumnos evaluadores y otro grupo de criterios se utilizarán en la evaluación realizada por los profesores. El peso de la evaluación de cada grupo (profesores y alumnos) se define también por los profesores creadores de las tareas.

En la evaluación, el usuario evaluador debe conceptualizar y hacer comentarios sobre cada uno de los criterios, y asignar una puntuación global (que no necesariamente debe de ser el promedio de los grados de los criterios) y un comentario general que debe resumir toda la evaluación. Los criterios que aparecen en la evaluación varían de acuerdo con los criterios estándares establecidos por los profesores. La figura 2 muestra la evaluación de una tarea.

Figura 2: Evaluación de una tarea

El profesor que creó la tarea es responsable por decidir hasta cuando puede recibir evaluaciones. En este momento, la nota final de la tarea se calcula automáticamente por el sistema basado en criterios establecidos previamente por el profesor, y el estado de la tarea pasa a 'cerrado'. Así, al final de cada semana, el sistema presentó una nota para cada alumno en la clase. Esta nota fue utilizada para calcular la calificación final del curso por el profesor.

A cualquier momento, los alumnos pueden consultar en el sistema el informe completo de evaluación de cada tarea, que exhibe la evaluación de cada evaluador en detalle: nota general, las observaciones generales y las notas y comentarios para cada criterio. En el sistema también es posible ver el informe del promedio de cada alumno por tarea, que permite acompañar el progreso de la clase.

4.2 Estudio de Caso 2: Evaluación de Conferencias

El Grupo de Usuarios Java de Río de Janeiro (Brasil) se reúne mensualmente para presentar y discutir la tecnología relacionada con la plataforma Java, mientras mantiene una lista de discusión online. Para programar una conferencia, el coordinador del grupo tiene la responsabilidad de elegir el orador y el tema de la Conferencia. En la fecha fijada, el orador celebra su presentación de acuerdo con su horario y el único registro de los presentes se realiza a través de una lista de presencia de papel.

A pesar del público participar activamente en la conferencia con preguntas y

observaciones, no existe un instrumento formal que les ofrezca la oportunidad de evaluar al orador y el discurso después de que termine. Una herramienta de evaluación para las conferencias posiblemente haría a los miembros de la comunidad más participativos en el grupo.

Por lo tanto, hay indicios de que la evaluación colaborativa de conferencias es una buena estrategia para el grupo RioJUG. Así, las evaluaciones de los usuarios de las conferencias pueden ser utilizadas como feedback para ayudar a los coordinadores en la selección de los oradores y temas futuros. Por otra parte, el rendimiento de los oradores que se registra en el sistema, puede actuar como histórico evolutivo para futuras consultas.

Fue utilizada una instancia del Framework IssueNet como herramienta para la evaluación colaborativa de las conferencias de RioJUG. En la nueva dinámica, además de determinar una fecha para la conferencia, el coordinador también ayuda con su evaluación, y los oyentes no sólo asisten a la conferencia, también registran sus impresiones a través de la evaluación. Así, todo el proceso deja de ser sólo una exposición y pasa a ser evolutivo, rectificado con las evaluaciones de las conferencias que los usuarios realizan.

Se identificaron dos perfiles de acceso diferentes para este estudio de caso: Los coordinadores, responsables por gestión de usuarios y perfiles, la creación y evaluación conferencias, y los Miembros, responsables por la presentación y evaluación de las conferencias. Las características del sistema varían según el perfil de acceso del usuario. La Figura 3 muestra la pantalla principal del sistema, en portugués.



Figura 3: Pantalla principal del sistema

Después que el coordinador registra la conferencia como una tarea, esta es resuelta por el conferencista en el momento de la presentación. Entonces, todos los espectadores pueden realizar sus propias evaluaciones. Para verificar el impacto que una forma de evaluación más o menos detallada causa en la experiencia de evaluación colaborativa, se decidió utilizar un modelo simple de evaluación en este estudio de caso: cada evaluador debe informar sólo una nota y un comentario opcional, a diferencia del estudio de caso presentado en la sección anterior, cuyo modelo de evaluación estaba formado por varios criterios. La operación de la evaluación de las conferencias de este estudio de caso se ilustra en la Figura 4.



Figura 4: Evaluación de una conferencia

Al final del período de evaluación de una conferencia, el coordinador creador debe cambiar su status a ‘cerrado’, para que no pueda recibir más evaluaciones. En este punto, el sistema permite consultar el informe completo de evaluación de la conferencia, con el promedio de las evaluaciones, todas las notas y los comentarios recibidos por cada uno de los evaluadores.

5. Discusión de los Experimentos

La sección anterior presentó dos estudios de caso realizados con el Framework IssueNet. En esta sección se presentan las opiniones de los participantes de los estudios de caso, y sus evaluaciones serán analizadas para identificar las declaraciones relacionadas con el objetivo de los estudios de caso.

5.1 Estudio de Caso 1: Evaluación de Trabajos Académicos

Los alumnos registraron sus impresiones sobre la utilización del sistema en la disciplina PSS a través de entrevistas individuales. Después de revisar las evaluaciones recibidas, se concluyó que la satisfacción de los alumnos con el sistema fue unánime: el 100% de los evaluadores clasificó el sistema como "bueno" o "excelente". Todos acordaron también que la calidad de la interfaz gráfica y facilidad de uso del sistema son buenos o excelentes, y la navegabilidad y la funcionalidad del sistema fueron clasificados como de “buena” o “excelente” por el 85% de los alumnos, y "regular" por otros 15%. La cantidad de problemas encontrados en el sistema no fue considerada impactante para la mayoría de los entrevistados (70%).

Cuando se les preguntó sobre el proceso de la evaluación colaborativa, la mayoría de los alumnos aprobó la experiencia. La mayoría de los alumnos (85%) respondió que, en comparación con la evaluación tradicional, en el que sólo el profesor evalúa, la evaluación colaborativa es buena o excelente, y también acordaron que evaluar sus colegas ayuda a mantener la atención en clase y aprender los conceptos teóricos de la disciplina. Mientras que el 70% de los evaluadores acordaron que evaluar a sus colegas ayuda a mejorar su propio trabajo, el 15% manifestó neutralidad. Cuando se les preguntó si evaluar sus colegas hace que el evaluador se sienta jugar un papel más importante, el 57% estuvo de acuerdo, el 15% fueron neutrales y el 28% estuvo contrario.

VIII CSPC - Proceedings, Huancayo 2009

Después de evaluar el estudio de caso, se puede concluir que tanto el sistema como la experiencia de la evaluación colaborativa fue bien recibida por la mayoría de los alumnos. Aunque haya recibido una excelente evaluación, el estudio de caso también recibió críticas,

especialmente con relación a la forma como fue conducido. Los testimonios indican que la evaluación colaborativa es considerada una experiencia interesante, pero que este proceso debe de estar bien planeado antes de su puesta en práctica. Además, muestra la importancia de la discusión entre los evaluadores durante y después del proceso de evaluación a fin de enriquecer el aprendizaje del grupo con el intercambio de ideas.

5.2 Estudio de Caso 2: Evaluación de Conferencias

A los participantes se les pidió registrar sus impresiones sobre el uso del sistema en la evaluación de las conferencias de RioJUG a través de entrevistas individuales. Este paso no sólo trató de recoger comentarios y sugerencias sobre el sistema, sino también de validar y evaluar el uso del Framework como una herramienta para el trabajo colaborativo.

Cuando se les preguntó acerca de la importancia de la evaluación de las conferencias, los participantes fueron unánimes: el 100% estaba de acuerdo que es importante evaluar, y el 77% señaló que la forma ideal de la evaluación es mediante la asignación de una puntuación a cada criterio de evaluación. En este caso, la metodología de evaluación del estudio consistió en la asignación de una sola nota y un comentario mediante conferencias, y estos datos indican que, para los usos futuros de esta instancia IssueNet en grupos de trabajo colaborativo, el proceso de evaluación se debe mejorar, al utilizar criterios de evaluación.

Cuando hicieron la evaluación de la conferencia, el 77% de los entrevistados llenaron el campo de comentarios. A los participantes que no lo llenaron se les preguntó por qué dejaron el campo en blanco, el 66% respondió que no tenía nada que agregar, mientras que otros (34%) dijeron que no respondieron porque no era obligatorio. Estos datos refuerzan la necesidad de cambio de paradigma que requiere la evaluación colaborativa, pues los evaluadores ya no tienen un papel pasivo, y ahora necesitan actuar activamente en el proceso de evaluación. Sobre el impacto que tiene el sistema de evaluación en la calidad de las conferencias, el 84% dijo que la evaluación ayuda a aumentar la calidad, frente al 7% que dijo que no contribuye pues, para contribuir, requeriría una evaluación más detallada de preferencia por criterios. Estos datos refuerzan la necesidad de adaptar este ejemplo para su uso en el trabajo colaborativo.

Casi la mitad de los entrevistados (46%) nunca fue conferencista, pero los que tienen el hábito de la presentación de conferencias, dijeron que desean recibir el feedback de sus conferencias: el 57% de ellos como un informe de evaluación detallado, mientras que el otro 43% prefiere un feedback rápido de la audiencia después de la presentación. Todos ellos se aprovechan de la crítica para mejorar en futuras presentaciones. Estos datos indican que la evaluación de las conferencias es muy importante para los conferencistas para mejorar la calidad de sus conferencias.

Después de evaluar el estudio de caso, se puede concluir que tanto el sistema como la experiencia de la evaluación colaborativa fue bien recibida por la mayoría de los participantes. Sin embargo, hubo muchas críticas sobre la metodología de valoración elegida: los testimonios demostraron que los participantes extrañaron los criterios de evaluación.

A partir de este estudio de caso se puede concluir que la evaluación colaborativa es interesante para los grupos de trabajo. Sin embargo, la metodología de evaluación debe de ser planeada cuidadosamente para que todos puedan beneficiarse: los evaluados, con los comentarios recibidos, y los evaluadores, con la mejora de la calidad de las conferencias.

6. Conclusiones

Esta pesquisa presenta un Framework para la evaluación colaborativa de tareas en grupos de trabajo CSDE-aprendizaje. Para validar la contribución del Framework en la evaluación colaborativa e identificar otras influencias sobre su uso en grupos de trabajo o de aprendizaje se llevaron a cabo dos estudios de caso con instancias del IssueNet. El primer estudio de caso buscó examinar la aplicación del Framework en grupos de aprendizaje. El segundo estudio de

caso tiene por objeto analizar su uso potencial para la evaluación de conferencias en grupos de trabajo.

Después de la finalización de los estudios de caso, se constató que el IssueNet fue bien aceptado, tanto en el grupo de aprendizaje como en el grupo de trabajo. Después de analizar las entrevistas, se concluyó que tanto el sistema como la experiencia de la evaluación colaborativa fue bien recibida por la mayoría de los participantes.

El primer estudio de caso indica que para los alumnos, la evaluación colaborativa es considerada una experiencia interesante, pero este proceso debe estar bien planificado antes de su puesta en práctica. Además, muestra la importancia de la discusión entre los evaluadores durante y después del proceso de evaluación a fin de enriquecer el aprendizaje del grupo con el intercambio de ideas. Desde el segundo caso de estudio, se concluyó que la evaluación colaborativa es interesante para los grupos de trabajo colaborativo, desde que la metodología de evaluación sea cuidadosamente planeada para que todos puedan beneficiarse: los evaluados, con los comentarios recibidos, y los evaluadores, con la mejora de la calidad de las conferencias.

Referencias

- Escovedo, T., Pimentel, M., Fuks, H. & Lucena, C.J.P., (2006). Avaliei, Avaliei...
Ensinei? Investigações Sobre Avaliação Nas Conferências de um Curso Online. XII
WIE - Workshop de Informática na Escola, Anais eletrônico do XXVI Congresso da
SBC. Campo Grande, MS, 14 a 20 de julho de 2006.
- Fayol, H., (1994). Administração industrial e geral. 10ª edição. Atlas, São Paulo.
- Ford, H., (1933). Minha vida e minha obra. Tradução de Monteiro Lobato. 3ª edição,
Companhia Editora Nacional, São Paulo.
- Fuks, H., Raposo, A. B. & Gerosa, M. A., (2003). Do Modelo de Colaboração 3C à
Engenharia de Groupware. Simpósio Brasileiro de Sistemas Multimídia e Web –
Webmidia 2003, Trilha especial de Trabalho Cooperativo Assistido por Computador,
03 a 06 de Novembro de 2003, Salvador - BA.
- Gamma, E., Helm, R., Johnson, R., Vlissides, J., (1995). Design Patterns: Elements of
Reusable Object-Oriented Software. Addison-Wesley.
- McConnell, D., (2002). Collaborative Assessment as a Learning Event in E-learning
Environments. CSCL 2002, Boulder, Colorado, Estados Unidos.
- Prata, D. N., (2003). Estratégias para o Desenvolvimento de um Framework de Avaliação da
Aprendizagem a Distância. In: XIV Simpósio Brasileiro de Informática na Educação, Rio
de Janeiro. Inclusão Digital como Instrumento de Inclusão Social. Rio de Janeiro :
Núcleo de Computação Eletrônica, 2003. p. 150-159.
- O'Sullivan, R.G., (2004). Practicing evaluation: a collaborative approach. California
SAGE, Califórnia, Estados Unidos.
- Taylor, F. W., (1990). Princípios de administração científica. 8ª edição. São Paulo: Atlas.

Extracción de Características en el Procesamiento Digital de una Señal para el Mejoramiento del Reconocimiento Automático de Habla usando Wavelets

Jorge Luis Guevara Díaz

Universidad Nacional de Trujillo, Escuela de Informática,
Trujillo, Perú
jorge.jorjasso@gmail.com

and

Juan Orlando Salazar Campos

Universidad Nacional de Trujillo, Escuela de Informática,
Trujillo, Perú
josc_orlando@hotmail.com

Resumen

La presente investigación hace un estudio en la aplicación de diversas wavelets y wavelets packets en el procesamiento digital de la señal para el reconocimiento automático del habla, se hacen diversos experimentos para medir el desempeño entre diversos tipos de wavelets, también se mide el desempeño de éstas frente a la técnica de MFCC que es una de las técnicas más robustas que existen; finalmente se mide el desempeño de las wavelets frente a la transformada de Fourier.

Se han experimentado con la transformada wavelet de Haar, Daubechies y Coiflets cuyos algoritmos tienen una complejidad computacional de $O(n)$, como también con los wavelets packets cuya complejidad computacional es de $O(n \log n)$, obteniendo mejores resultados para los wavelets packets de Daubechies 6, haciendo un análisis aproximado de las frecuencias tal como lo hace la escala Mel.

Palabras clave: Muestreo, Wavelets, cepstrum, Mel, Fourier, Dynamic Time Warping, Procesamiento Digital de Señales.

1. Introducción

La presente investigación esta centrada en el Reconocimiento Automático del Habla (RAH) y mas concretamente en el uso de la transformada Wavelet en la parte del procesamiento digital de la señal y su influencia en la extracción de características.

Las Wavelets han sido utilizadas ampliamente en diversos campos como procesamiento de imágenes y señales, física, matemática incluso economía; los algoritmos rápidos de bajo costo computacional es lo que las hace tan atractivas. En el presente trabajo se diseñó un algoritmo basado en características perceptuales inspirados en los Coeficientes Cepstrales en Escala Mel (MFCC), mediante wavelets packets, también se experimentó con diversos tipos de wavelets discretos.

El resto de éste paper está organizado de la siguiente manera. En la sección 2 se muestra algunos trabajos previos realizados, la sección 3 describe el método MFCC para extracción de características. Se describen las wavelets, además, se trata la Transformada Wavelet frente a la Transformada de Fourier en la sección 4. En la sección 5 se muestra el modelo propuesto por este trabajo de investigación en el uso de la Transformada Wavelet en el Reconocimiento Automático del Habla. En la sección 6 se muestra el modelo propuesto en el uso de las wavelets Packets. En la sección 7 van los experimentos y resultados. En la sección 8 se encuentran las conclusiones del presente trabajo. Finalmente en la sección 9 se analizan futuras investigaciones que pueden surgir del presente trabajo de investigación.

2. Trabajos Previos

Existen diversos trabajos realizados, entre los más importantes detallaremos a continuación los siguientes. En [13], se describen diversos métodos de procesamiento digital de la señal, pero sin involucrar el uso de las wavelets En [1] describe a manera de introducción la posible aplicación de los Wavelets en el RAH, los trabajos de sarikaya, [16], [15], son unos de los mas importantes en lo que se refiere a la aplicación de Wavelets en la parte del procesamiento digital de la señal para el reconocimiento del Hablante, entre otros trabajos no menos importantes, tenemos [17], que hace uso de la transformada Wavelet en el reconocimiento de fonemas y el trabajo de, [10], que hace uso de la aplicación de Wavelets también en el reconocimiento del hablante. En la parte de emparejamiento de patrones uno de los trabajos mas importantes es el trabajo de Sakoe and Chiba, [14] que muestran un algoritmo optimizado para el reconocimiento de palabras haciendo uso de la programación dinámica. Una de las técnicas más usadas hoy en día son los Modelos Ocultos de Markov [3], existen también un sinnúmero de trabajos en Redes Neuronales , Máquinas de Soporte Vectorial aplicadas en la etapa de reconocimiento. Otro trabajo importante es el de [9] quien hace un descripción muy amplia de muchos algoritmos utilizados y una descripción muy detallada de los sistemas informáticos del lenguaje hablado, incluyendo temas como reconocimiento automático del habla, síntesis del habla y entendimiento del lenguaje natural.

Por el lado de las wavelets éstas han sido introducidos recientemente a principios de los años ochenta y han llegado a ser de gran interés en diversas disciplinas, pero sus raíces datan de mucho tiempo atrás. En la actualidad las wavelets han tomado una enorme popularidad. Sin embargo, sus raíces datan de 1873, cuando el trabajo de Karl Weierstrass describió una familia de funciones que son construidas por una superposición de copias escaladas de una función base dada. Trabajos importantes son los de Haar, el trabajo de Dennis Gabor quién describió una base no-ortogonal de lo que ahora se llaman wavelets con soporte no acotado, basado en funciones gaussianas trasladadas, También se deben citar diversos trabajos de Daubechies[4], Morlet y Grossmann [5], Mallat[12], Yves Meyer y muchos otros investigadores quienes han aportado mucho al desarrollo de este campo de estudio.

3. Procesamiento de la señal

La extracción de características de manera tradicional se hace utilizando algunas técnicas, las mas usadas son los coeficientes de predicción lineal Cepstrales LPC-cepstrum [2], coeficientes de predicción lineal perceptuales PLP [8], los coeficientes MFCC.

En el presente trabajo haremos nuestra comparación con la técnica de MFCC, pues esta técnica está basada en un modelo de percepción del habla, y la técnica a mostrar utiliza Wavelet y también esta basada en la percepción humana del habla, como también compararemos los resultados obtenidos mediante el uso de las Wavelets frente al uso de la Transformada de Fourier calculando las energías de diversos bandos de frecuencias.

3.1. Coeficientes MFCC

La señal de voz una vez capturada y digitalizada, es segmentada en frames muy pequeños según algunos experimentos neurofisiológicos en la codificación del habla sugieren que esta segmentación debe ocurrir alrededor de 10 mS, pues alrededor de 20 mS el oído empieza a oír una cierta distorsión,

$$\chi^m[n] = \chi[n - mF]\omega[n]. \quad (1)$$

donde "m" es el frame a procesarse F es el espaciamiento entre frames y $\omega[n]$, es una ventana de longitud N, esta ventana puede ser rectangular o del tipo hamming o hanning, una ventana rectangular produce grandes ondas laterales, y da el máximo ajuste la ventana hamming no tiene tanta precisión frecuencial peso provoca efectos mucho menores

$$\omega[n] = 0,54 - 0,46 \cos \frac{2\Pi n}{N} \text{Hamming} \quad (2)$$

luego una trasformada discreta de fourier es aplicada a cada frame para obtener los componentes de frecuencia de la señal, generalmente se usa un algoritmo que implemente de manera eficiente la Trasmformada Rápida de Fourier

$$X_m(e^{jw}) = \sum \chi_m[n]e^{-jw} = \sum \omega[m - n]\chi[n]e^{-jw} \quad (3)$$

una vez llevado al dominio de la frecuencia cada frame es pasado a la escala Mel , la escala Mel es una escala construida en base a la percepción humana del habla, y sus valores han sido dados tras experimentos fisiológicos de muchos investigadores quienes han construido escalas de frecuencias basadas en la respuesta natural del sistema de audición humano, pues el complejo sistema auditivo trata las frecuencias de entrada en una manera no lineal , sino mas bien de una manera casi logarítmica, la escala en frecuencia Mel esta dada por:

$$\beta(f) = 1125 \ln\left(1 + \frac{f}{700}\right) \quad (4)$$

y para traspasar las frecuencias obtenidas por medio del Algoritmo Rápido de la Trasmformada Discreta de Fourier se procede hacer un ventanamiento llamado "bins" de la siguiente manera:

$$H = \begin{cases} 0 & \text{si } k < f(m-1) \\ \frac{k-f(m-1)}{f(m)-f(m-1)} & f(m-1) \leq k \leq f(m) \\ \frac{k-f(m-1)}{f(m+1)-f(m)} & f(m) \leq k \leq f(m+1) \\ 0 & k > f(m+1) \end{cases} \quad (5)$$

donde

$$f(m) = \frac{N}{F_s} \beta^{-1}\left(\beta(f_1) + m \frac{\beta(f_h) - \beta(f_1)}{M+1}\right) \quad (6)$$

una vez obtenidas las frecuencias bineadas en escala Mel , se procede a calcular el cepstrum cuya justificación viene del hecho que se tiene el modelo de fuente de voz dice que el timbre y tonalidad de la señal de voz estan en convolución con el filtro del tracto vocal, las partes altas en el tiempo del cepstrum viene a ser el timbre o inflexiones de voz propias de cada persona al hablar , y las partes bajas corresponde a la información proveniente del tracto vocal.

La salida del filtro para cada frame esta dado como sigue:

$$S(m) = \ln\left(\sum |X(k)|H_m(k)\right), 0 < m < M \quad (7)$$

donde X(k) es la salida de la Trasmformada Discreta de Fourier.

Finalmente una Trasmformada Discreta del Coseno-II es Calculada,por concentrar su energía alrededor de las frecuencias mas bajas.

$$c(m) = \sum S(m) \cos\left(\pi n \left(\frac{m + \frac{1}{2}}{M}\right)\right) \quad (8)$$

Estos coeficientes resultantes son los vectores de características resultantes de la parte del procesamiento digital de la señal , estos coeficientes toman el nombre de Coeficientes Cepstrales en Escala Mel MFCC, nombre obtenido por el procedimiento antes mencionado.

4. Transformada Wavelet

La Transformada Wavelet es una herramienta matemática que corta los datos, funciones o operadores en diferentes componentes de frecuencia [4] y estudia cada componente a una resolución ubicada a esa escala.

4.1. Transformada Wavelet Continua

Restringiendo a una dimensión y estableciendo los parámetros de dilatación y traslación a y b que varían continuamente sobre \Re con la restricción de $a \neq 0$, la transformada wavelet continua de una función f está dada por:

$$\begin{aligned} (T^{wav} f)(a, b) &= \int \delta x f(x) |a|^{-\frac{1}{2}} \psi \\ (T^{wav} f)(a, b) &= \langle f, \psi^{a,b} \rangle \left(\frac{x-b}{a} \right) \end{aligned} \quad (9)$$

la familia de wavelets se puede construir dilatando y trasladando

$$\psi^{a,b}(x) = |a|^{-\frac{1}{2}} \psi\left(\frac{x-b}{a}\right) \quad (10)$$

la función f puede ser recuperada de su transformada wavelet como sigue:

$$f = C_{\psi}^{-1} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \frac{\delta a \delta b}{a^2} \psi(T^{wav} f)(a, b) \psi^{a,b} \quad (11)$$

4.2. Comparación de la Transformada de Fourier con la Transformada Wavelet

A continuación se muestran las diferencias y similitudes del análisis Wavelet frente al análisis con Fourier.

$$T^{win}(w, t) = \int \delta s f(s) g(s-t) e^{-i\omega s} \quad (12)$$

Transformada Ventaneada de Fourier.

$$CWT(a, b) = \frac{1}{\sqrt{a}} \int f(x) \psi\left(\frac{x-b}{a}\right) \delta x \quad (13)$$

Transformada Wavelet.

La transformada Wavelet provee una descripción similar Tiempo - Frecuencia. Una similitud entre la Transformada Wavelet y la Transformada Ventaneada de Fourier sería en que ambas toman el producto interno de la función f con una familia de funciones $g(s-t)e^{-i\omega s}$ y con $\psi\left(\frac{x-b}{a}\right)$, donde las funciones $\psi^{a,b}$ son llamadas Wavelets.

La diferencia entre la Transformada Wavelet y la Transformada Ventaneada de Fourier está dada en el hecho de la manera en como analizan las funciones [4], la función g analiza utilizando la misma forma para las frecuencias altas y las frecuencias bajas, la función ψ analiza las altas frecuencias con pequeñas formas y las bajas frecuencias con tamaño mucho mayores.

Para más detalle sobre Wavelets ver [4]

5. Extracción de características utilizando Wavelets

De la misma manera que los Coeficientes Cepstrales en Escala Mel hacen un filtrado de los componentes de frecuencia, y luego una decorrelación del cepstrum mediante una transformada del coseno, la extracción de características en el procesamiento digital de la señal mediante wavelets se procedió a hacer algo similar.

Primero se hizo una descomposición wavelet de cada frame hasta j niveles que corresponderán a un análisis multiresolución, donde la señal es proyectada a cada nivel de resolución obteniendo al final j espacios W correspondientes diversos rangos de frecuencia y un espacio V correspondiente al nivel mas bajo de frecuencia de la señal, para esto se utilizó las wavelets discretas:

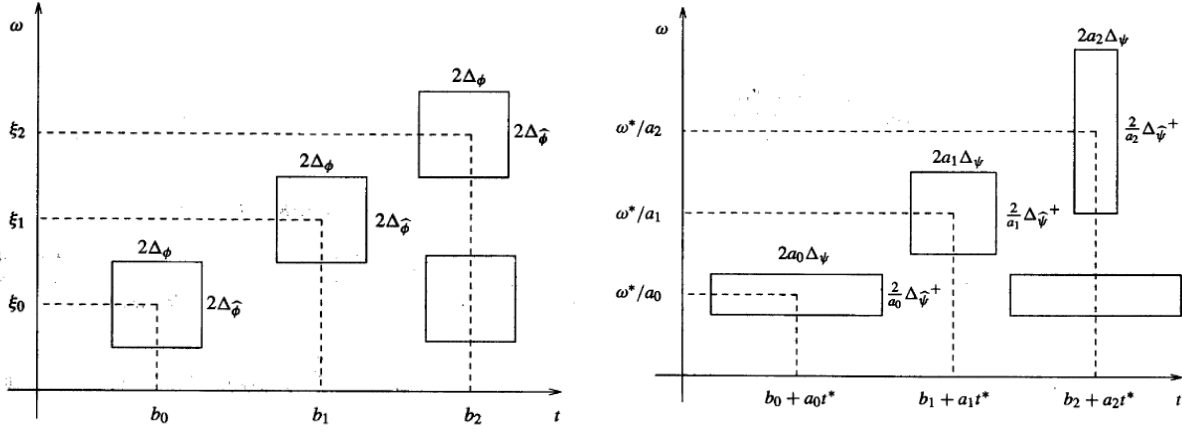


Figura 1: Ventana Tiempo-Frecuencia para la Transformada de Fourier y para la Transformada Wavelet. Fuente: [11]

$$\begin{aligned}\psi^{m,n}(x) &= a_0^{-\frac{m}{2}} \psi(a_0^{-m}(x - nb_0 a_0^m)) \\ \psi^{m,n}(x) &= a_0^{-\frac{m}{2}} \psi(a_0^{-m}x - nb_0)\end{aligned}\quad (14)$$

en particular para este trabajo se eligió $a_0 = 2$ y $b_0 = 1$ entonces:

$$\psi^{m,n}(x) = 2^{-\frac{m}{2}} \psi(2^{-m}x - n) \quad (15)$$

En el espacio V y en cada espacio W , tendremos coeficientes escala ϕ y coeficientes wavelet ψ correspondientes, en este caso solo se utilizarán los coeficientes $C_{j,k}$ y $d_{j,k}$ para determinar información importante en determinado espacio de tiempo en la señal de habla, coeficientes con altos valores nos indicarán la presencia de información importante.

Para proceder a obtener la extracción de características mediante wavelets, se hizo un ventaneamiento (windowing) de la señal utilizando una ventana Hamming de $16mS$ para ciertos experimentos y también ventanas Hamming de tamaño $32mS$, también se utilizó un tamaño de paso F de $10mS$. el tamaño de escogido de esas ventanas y valores de paso fué experimental y con ellos se lograron mejores resultados

Una vez obtenidos los valores del ventaneamiento se procedió a hacer una descomposición wavelet de cada segmento obtenido del ventaneamiento, hasta un nivel $j=7$ para las wavelets de haar $j=6$ para las wavelets de Daubechies 4, $j=5$ para las wavelets de Daubechies 6 y Coiflets 6. El proceso de descomposición utilizó el algoritmo de banco de filtros

5.1. Algoritmo de banco de filtros

Desde que $\phi \in V_0 \subset V_{-1}$ y que $\phi_{-1,n}$ son bases ortonormales en V_{-1} , tenemos:

$$\phi(x) = \sqrt{2} \sum_n h_n \phi(2x - n) \quad \text{con} \quad h_n = \langle \phi, \phi_{-1,n} \rangle \quad \text{y} \quad \sum_{n \in \mathbb{Z}} |h_n|^2 = 1 \quad (16)$$

esto indica que la función escala en cierta nivel m puede ser expresada en términos de funciones escalas trasladadas en la siguiente escala mas pequeña.

Similarmente podemos expresar la función wavelet en cierto nivel en términos de funciones escaladas y trasladadas en la siguiente escala mas pequeña.

$$\psi(x) = \sqrt{2} \sum_n g_n \phi(2x - n) \quad \text{con} \quad g_n = \langle \psi, \phi_{-1,n} \rangle = (-1)^n h_{-n+1} \quad (17)$$

lo cual implica los dos importantes resultados:

Si una función f puede ser representada por funciones escala en el nivel m

$$f(t) = \sum_n C_n \phi_{-1,n} \quad \text{con} \quad C_n = \langle f, \phi_{-1,n} \rangle \quad (18)$$

y en términos de wavelets

$$f(t) = \sum_n d_n \psi_{-1,n} \quad \text{con} \quad d_n = \langle f, \psi_{-1,n} \rangle \quad (19)$$

finalmente se puede establecer lo siguiente:

$$C_m = \sum_l h(l - 2n) C_{m-1}(l) \quad , \quad d_m = \sum_l g(l - 2n) d_{m-1}(l) \quad (20)$$

Estas dos ultimas ecuaciones nos dicen que los coeficientes wavelets y escala en cierto nivel m pueden ser encontrados de manera iterativa, por ejemplo empezando de $\langle f, \phi_{0,n} \rangle$ podremos calcular $\langle f, \psi_{1,n} \rangle$ y $\langle f, \phi_{1,n} \rangle$ y así sucesivamente.

Los valores para h y g actúan como filtros de paso baja y filtros de paso alto respectivamente y llamaremos a h el filtro escala y a g el filtro wavelet.

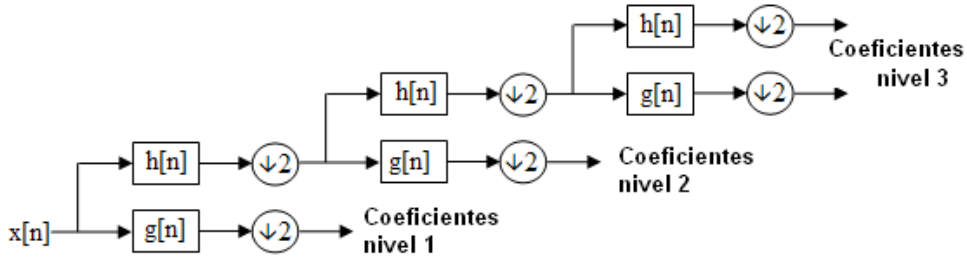


Figura 2: Implementación del banco de filtros iterativo. Fuente [7]

5.2. Complejidad Computacional del algoritmo de banco de filtros

La complejidad computacional de este algoritmo es la siguiente:

$$T[n] = \begin{cases} T(\frac{n}{2}) + Cn & \text{si } 2^n \geq 2 \\ 0 & n = 1 \end{cases} \quad (21)$$

resolviendo la ecuación de recurrencia se tiene que este algoritmo para la transformada wavelet con banco de filtros tiene una complejidad de $O(n)$. y para el caso de las wavelets packet este algoritmo tiene una complejidad de $O(n \log n)$, mas detalle en [7]

6. Modelo propuesto para la Extracción de Características basadas en Wavelets Packets Perceptuales

En este modelo que proponemos hacemos uso de los wavelet Packet, pero no nos interesa toda la descomposición del wavelet packet, solo deseamos obtener los coeficientes que están en determinado nivel de resolución, cuyos componentes de frecuencia son aproximadamente iguales a la escala Mel.

Para este caso utilizamos un tamaño de frame igual a $24mS$ y un tamaño de paso igual que el caso anterior de $16mS$.

6.1. Obtención de las Características

Una vez computados los valores de los coeficientes para cada nivel de descomposición se procedió a calcular la energía [10] de cada nivel con la finalidad de comprender el aporte del nivel en el tiempo, la energia de cada nivel se calculó mediante la siguiente expresión.

$$E_i = \frac{\sum_{j=1}^N (W_i^P f(j))^2}{N_i} \quad \text{VIII CSPC - Proceedings, Huancayo 2009} \quad (22)$$

Arbol de descomposición.

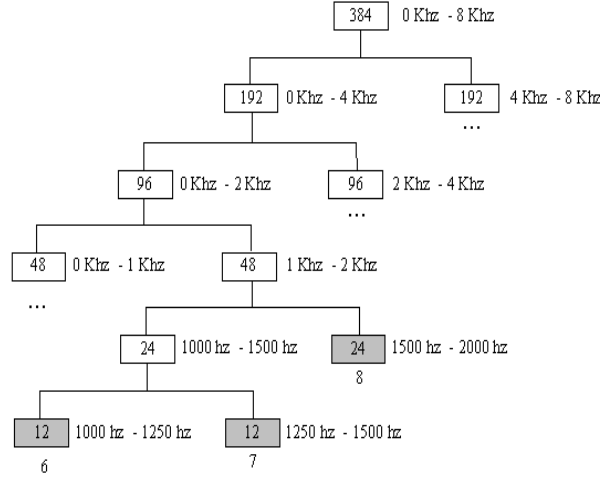


Figura 3: Arbol de descomposición espacios de resolución 6,7 y 8 para las wavelet packet. Fuente [7]

Donde N_i es el número de coeficientes wavelet en el nodo i , $W_i^P f(j)$ es el j th coeficiente de las wavelets packets de una señal f en el nodo W_i^P , P es el total de nodos usados, finalmente se aplica una transformada discreta del coseno al logaritmo de las energías para cada bando de frecuencia, estos valores son los que constituirán los vectores de características de la señal

$$F(i) = \sum_{i=1}^P \log E_n \cos\left(\frac{i(n-1)}{P}\right) \quad (23)$$

7. Experimentos y Resultados

Se realizaron varios experimentos con varios tipos de wavelets que a continuación se detallan. Wavelet de Haar

$$h(n) = \left[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right] \quad (24)$$

Wavelets de Daubechies 4

$$h(n) = \left[\frac{1 + \sqrt{3}}{4\sqrt{2}}, \frac{3 + 3\sqrt{3}}{4\sqrt{2}}, \frac{3 - 3\sqrt{3}}{4\sqrt{2}}, \frac{1 - \sqrt{3}}{4\sqrt{2}} \right] \quad (25)$$

Wavelets de Daubechies 6

$$h(n) = [0,3326, 0,8068, 0,4598, -0,1350, -0,0854, 0,0352] \quad (26)$$

Wavelet Coiflet 6

$$h(n) = \left[\frac{1 - \sqrt{7}}{16\sqrt{2}}, \frac{5 + \sqrt{7}}{16\sqrt{2}}, \frac{14 + 2\sqrt{7}}{16\sqrt{2}}, \frac{14 - 2\sqrt{7}}{16\sqrt{2}}, \frac{1 - \sqrt{7}}{16\sqrt{2}}, \frac{-3 + \sqrt{7}}{16\sqrt{2}} \right] \quad (27)$$

Siendo los filtros g correspondientes: $g_n = (-1)^n h_{-n+1}$

El tipo de pruebas que se realizaron fueron independiente del hablante, tomando como plantillas para el reconecedor 125 palabras de entrenamiento pertenecientes a 5 personas. (25 palabras por persona). Para los casos de test se utilizaron 800 palabras (15 palabras por 60 personas) por método implementado. haciendo un total de 8100 pruebas realizadas; las palabras utilizadas fueron: abajo, abrir, adiós, arriba, caminar, cancelar, cerrar, coger, cuatro, derecha, detener, dos, eliminar, error, guardar, hola, iniciar, izquierda, pez, salir, terminar, test, tres, tres(repetición), uno. Para las pruebas se utilizaron 15 palabras por persona, las palabras fueron: abajo, abrir, adiós, arriba, caminar, cancelar, error, hola, izquierda, pez, salir, terminar, tres, tres

(repetición), uno. Para realizar las pruebas desarrollamos el software LORITO [6] en el que se implementaron todos los algoritmos estudiados y gracias al cual pudimos efectuar nuestras tomas de muestras y el respectivo procesamiento de las mismas.

El método estadístico que nos permitió analizar nuestro resultado fue la Prueba JI Cuadrado de MC Nemar-Datos Correlacionados

<i>Método</i>	<i>Tasa aceptación</i>	<i>Error</i>
Coeficientes Cepstrales en Escala Mel	85.32%	14.68%
Wavelet Haar	34.47%	65.53%
Wavelet Daubechies 4	51.79%	48.21%
Wavelet Daubechies 6	61.32 %	38.68%
Wavelet Coiflets 6	55.46 %	44.54%
Wavelet Packet Perceptuales Walsh	55.79 %	44.21 %
Wavelet Packet Perceptuales Daubechies 4	69.14 %	30.86%
Wavelet Packet Perceptuales Daubechies 6	74.43 %	25.57%
Wavelet Packet Perceptuales Daubechies 4 (22)	71.21 %	28.79%

Figura 4: Datos obtenidos utilizando la técnica de DTW como reconocedor, con distancia Chebyshev y con Slope Constrain P=1. Se observa la mejor performance en las Wavelet Packet Perceptuales Daubechies 6, y la mas pobre en las Wavelet Haar. Datos obtenidos con software LORITO [6]

	<i>MFCC</i>	<i>W. Haar</i>	<i>W. Db4</i>	<i>W. Db6</i>	<i>W. Coif6</i>	<i>WP Walsh</i>	<i>WP Db4</i>	<i>WP Db 6</i>	<i>WP Perc.</i>
<i>Arriba</i>	86%	17%	41%	48%	34%	31%	66%	72%	62
<i>Cerrar</i>	100%	45%	79%	86%	76%	97%	100%	97%	97%
<i>Coger</i>	90%	17%	17%	31%	17%	28%	52%	69%	62%
<i>Cuatro</i>	97%	48%	28%	31%	24%	62%	76%	83%	83%
<i>Dos</i>	86%	28%	38%	48%	48%	59%	69%	69%	72%
<i>Eliminar</i>	72%	24%	34%	52%	48%	24%	48%	76%	45%
<i>Error</i>	97%	24%	66%	93%	83%	93%	97%	97%	93%
<i>Hola</i>	83%	38%	48%	72%	55%	38%	52%	59%	52%
<i>Izquierda</i>	86%	31%	48%	62%	59%	24%	59%	72%	69%
<i>Pez</i>	62%	24%	52%	41%	66%	62%	59%	59%	55%
<i>Salir</i>	86%	66%	79%	79%	72%	76%	76%	79%	76%
<i>Terminar</i>	62%	24%	24%	31%	28%	41%	55%	55%	55%
<i>Tres</i>	83%	24%	52%	59%	45%	41%	66%	62%	66%
<i>Tres</i>	59%	24%	45%	59%	41%	48%	55%	59%	62%
<i>Uno</i>	76%	31%	79%	83%	86%	76%	72%	66%	59%

Figura 5: Tasa de reconocimiento de las palabras por método. Datos obtenidos con software LORITO [6]

8. Conclusiones y futura investigación

El mejoramiento del espectro se da gracias al análisis tiempo frecuencia de las wavelets, con las wavelets podemos saber aproximadamente el aporte de las frecuencias por nivel de tiempo en las señales de habla, pues analiza con pequeñas wavelets componentes de alta frecuencia en la señal y con wavelets mas grandes componentes de baja frecuencia presentes en la señal, esto se traduce en la tasa de reconocimientos que proporcionan los wavelets. Una extracción de características usando solamente la Transformada de Fourier no muestra buenos resultados, pues ésta hace un análisis tiempo frecuencia de la señal con ventanas del mismo tamaño para todos los niveles de frecuencia sacrificando resolución de tiempo o frecuencia según se empequeñezca o agrande la ventana de análisis empobreciendo de esta manera la resolución espectral o temporal. La efectividad de reconocimiento de habla del MFCC radica en que es un buen modelo de

representación de la producción y percepción de habla, el cual es obtenido gracias a la agrupación de diversos métodos como el cepstrum, la escala Mel, Transformada de Fourier, etc, el mismo hecho de agrupar varios métodos para obtener mayor efectividad en el reconocimiento, eleva el tiempo de ejecución del algoritmo, llegando a obtener una complejidad tiempo de $O(n \log n)$, por frame de tamaño n , y un tiempo de ejecución mucho mayor. Las wavelets pueden ser utilizados alternativamente, para el procesamiento digital de la señal de habla. aprovechando el análisis que permiten y su rápida implementación computacional. Las wavelets discretas implementadas con el algoritmo de banco de filtros, para la extracción de características brindan una tasa de reconocimiento bajo, por lo cual se recurren a las wavelets packets que tienen mayor resolución en frecuencias, a las cuales hemos adaptado de tal manera de que en los espacios de resolución las wavelets tengan una frecuencia aproximada a la de la escala Mel. La complejidad computacional de los algoritmos de extracción de características usando las wavelets y las wavelets packets es de $O(n)$ y de $O(n \log n)$ respectivamente. La complejidad computacional de los MFCC y de las wavelets packet es la misma $O(n \log n)$, pero el menor tiempo de ejecución corresponde a las wavelets packet, debido al menor número de procedimientos utilizados. La ventaja de utilizar wavelets radica, en la variedad de funciones wavelet que se puede escoger, además de sus formas discretas y continuas.

Se pueden utilizar otra gamma de wavelets teniendo en consideración los resultados obtenidos en el presente trabajo, como también el uso de wavelets continuos. Se puede también segmentar la señal de una manera no uniforme, con tamaños variables de frames obtenidos gracias a las variaciones locales en la señal. La tasa de reconocimiento puede variar según el reconocedor que se use, en este caso hemos hecho uso de un reconocedor basado en un algoritmo optimizado de programación dinámica DTW, pero puede utilizarse otros reconocedores basados en Redes Neuronales, uso de probabilidades, Modelos Ocultos de Markov, Redes Bayesianas, Máquinas de Soporte Vectorial, etc. La tasa de reconocimiento que se ha mostrado en los métodos implementados se ha obtenido construyendo un reconocedor del tipo independiente del hablante, esta tasa puede variar muy notablemente en un sistema dependiente del hablante, donde es de esperarse tasas de reconocimientos mucho mayores.

Referencias

- [1] ABOUFADEL, E. A wavelets approach to voice recognition. *Grand Valley State University* (2001).
- [2] ATAL, AND SCHROEDER. Predictive coding of speech signals. *Report of the 6th Int. Congress on Acoustics, Tokyo, Japan* (1968).
- [3] BAUN, L., AND EAGON, J. Perceptual linear predictive analysis of speech. *RBulletin of American Mathematical Society, 1967, 73, pp. 360-363* (1968).
- [4] DAUBECHIES, I. *Ten lectures on wavelets*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1992.
- [5] GROSSMANN, A., AND MORLET, J. Decomposition of hardy functions into square integrable wavelets of constant shape. *SIAM Journal on Mathematical Analysis* 15, 4 (1984), 723–736.
- [6] GUEVARA, J. L. *Lorito, speech recognition software*, v 1.0 ed. Universidad Nacional de Trujillo, Trujillo, Enero 2007.
- [7] GUEVARA, J. L., AND SALAZAR, J. O. *Extracción de Características en el Procesamiento Digital de una Señal para el Mejoramiento del Reconocimiento Automático de Habla usando Wavelets*. Tesis, Trujillo, Enero 2007.
- [8] HERMANSKY, H. A an inequality with applications to statistical estimation for probabilistic functions of markov processes and to a model for ecology. *J. Acoust. Soc. Am.* (2005).
- [9] HUANG, X., ACERO, A., AND HON, H.-W. *Spoken Language Processing: A Guide to Theory, Algorithm and System Development*. Prentice Hall PTR, April 2001.
- [10] M. SIAFARIKAS, TODOR GANCHEV, N. F. Objective wavelet packet features for speaker verication, 2000.
- [11] MALLAT, J. Multiresolution approximation and wavelets. *Trans. Amer. Math. Soc.*, 315, pp. 69-68. (1989). 47

- [12] MALLAT, S. G. A theory for multiresolution signal decomposition: the wavelet representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 11, 7 (1989), 674–693.
- [13] MANTHA, V., R. Y. A. J. Implementation and analysis of speech recognition frontends.
- [14] SAKOE, H., AND CHIBA, S. Dynamic programming algorithm optimization for spoken word recognition. 159–165.
- [15] SARIKAYA, R., AND HANSEN, J. High resolution speech features parametrization for monophone based stressed speech recognition packet transform features with application to speaker identification., 2000.
- [16] SARIKAYA, R., PELLOM, B., AND HANSEN, J. Wavelet packet transform features with application to speaker identification, 1998.
- [17] TAN, B. T., FU, M., SPRAY, A., AND DERMODY, P. The use of wavelet transforms in phoneme recognition. In *Proc. ICSLP '96* (Philadelphia, PA, 1996), vol. 4, pp. 2431–2434.

Apéndice

Imágenes de LORITO [6] Software desarrollado para realizar los experimentos

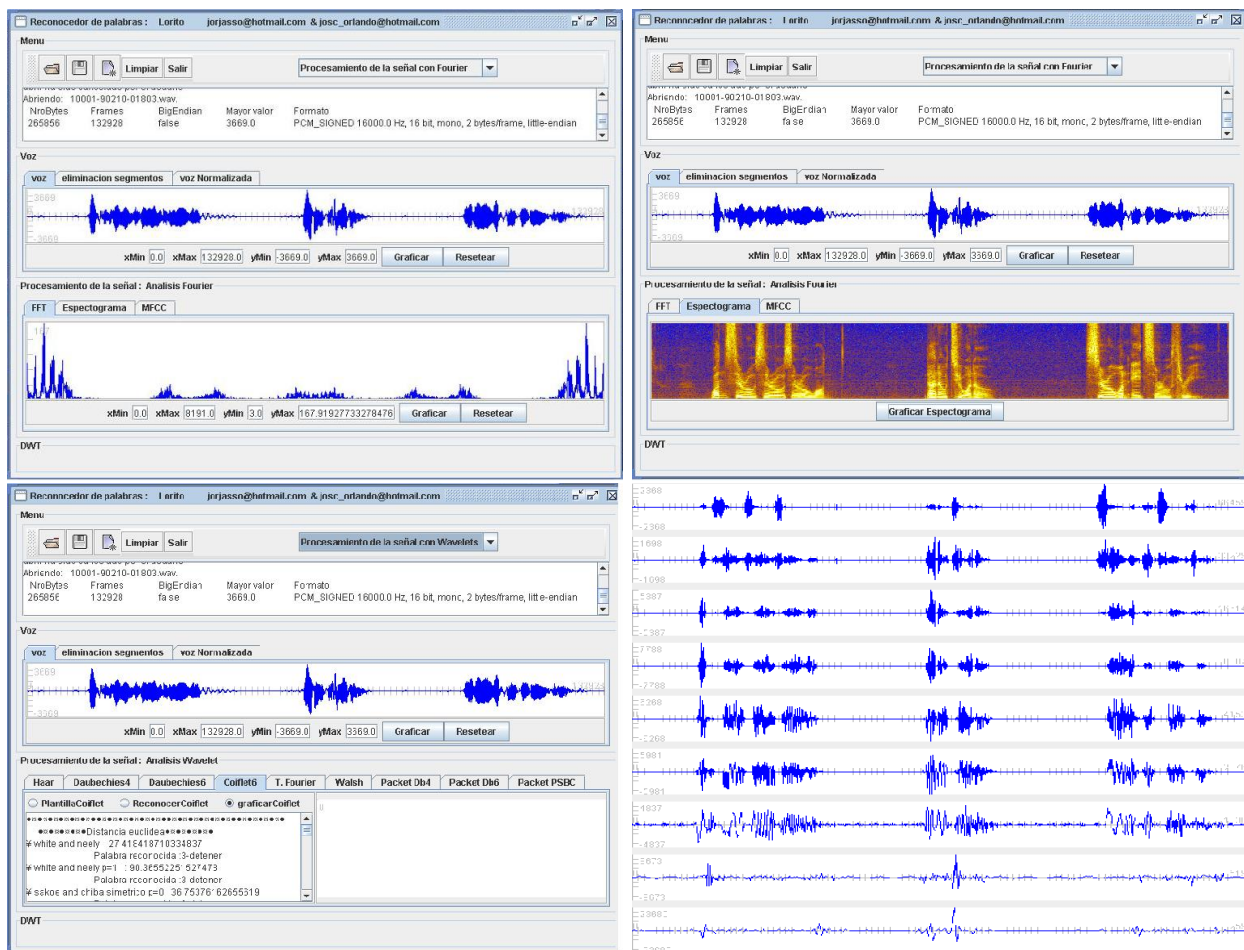


Figura 6: LORITO [6] en acción, Trasformada de Fourier de una palabra: parte superior izquierda, Espectrograma: parte superior derecha, Reconociendo una palabra: parte inferior izquierda, Análisis mediante Wavelets: parte inferior derecha

VIII Peruvian Computing Week JPC 2009

Noviembre 28-29, Huancayo - Peru

Gonzalo Manuel Chevarría Lock, Luis Atoche Castromonte, Asesora: Ing. Waldy Grández Pizarro

Universidad de San Martín de Porres
Facultad de Ingeniería y Arquitectura
Escuela Profesional de Ingeniería de Computación y Sistemas

gonzalochev@gmail.com, atochec@gmail.com, wgrandezp@usmp.edu.pe

IMPLEMENTACIÓN DE SERVIDOR XS PARA DESPLIEGUE DE PROYECTO OLPC EN ESCUELAS DEL PERÚ

Resumen

El proyecto que presentamos a continuación tiene su origen en una iniciativa de algunos alumnos de la Escuela Profesional de Ingeniería de Computación y Sistemas de la Facultad de Ingeniería y Arquitectura de la Universidad de San Martín de Porres la cual consiste en la interacción de un servidor escuela o servidor XS con un servidor principal ubicado en las instalaciones de la FIA USMP; con la finalidad de obtener reportes específicos de temas relevantes que sirvan como indicadores para aportar al Proyecto OLPC (Una Laptop por Niño) en las escuelas en las cuales se ha desplegado en nuestro país.

Palabras Claves: Server XS, OLPC, XO laptop

Abstract

The project presented below starts with the initiative of some students from the professional career of system engineering; Faculty of Engineering and Architecture of the San Martín de Porres University, which consists in the interaction of a school server XS or another principal server which is located in the FIA – USMP faculty, to obtain specific reports of relevant issues that serve as main purpose, this indicators will be use to contribute to the OLPC project (One Laptop per Child) which have been deployed in schools around our country.

Keywords: Server XS, OLPC, XO Laptop

1. INTRODUCCION

El desarrollo del presente trabajo nace por la problemática identificada en la integración y comunicación entre los centros educativos donde se ha desplegado las laptops XO, para la identificación del problema, sus causas, efectos, los medios y fines, se utilizó la metodología de Marco Lógico la cual se traduce en el árbol de problemas y el árbol de objetivos.

El análisis de los objetivos permite describir la situación futura a la que se desea llegar una vez se han resuelto los problemas.

ARBOL DE PROBLEMAS



ARBOL DE OBJETIVOS



OBJETIVOS

Objetivo General:

- Integración y comunicación entre los centros educativos donde ha sido desplegado el proyecto OLPC (una laptop por niño) y un servidor principal.

Objetivos Específicos:

- Controlar el uso de las laptops XO en los puntos desplegados.
- Compartir experiencias educativas entre profesores y alumnos de los diferentes puntos de nuestro país.
- Establecer un canal de comunicación formal para hacer el reporte de las de las fallas de las laptops XO.

Figura N° 1: Primera etapa del despliegue del proyecto OLPC en Perú



Fuente: www.perueduca.edu.pe/olpc

2. OTRAS SECCIONES:

descripcion del proyecto

El presente proyecto consta de tres etapas. La primera etapa consiste en el reconocimiento del Escenario, la segunda etapa es configurar y personalizar el “servidor escuela” y la tercera es la implementación del servidor principal en la FIA-USMP el cual interactuara con el servidor de la escuela piloto.

A continuación se muestra la Tabla que especifica cuantas laptops soporta un servidor dependiendo de la capacidad del procesador, lo cual servirá como referencia para el presente proyecto.

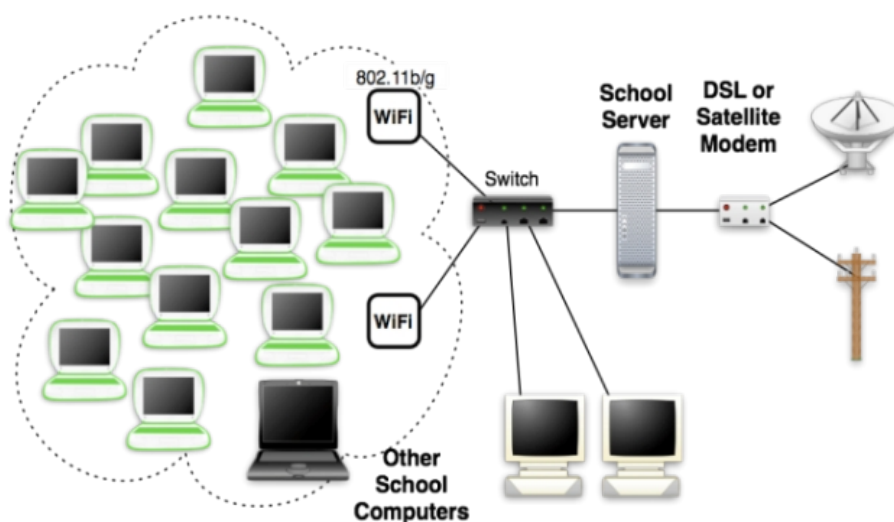
Tabla N°1: Recomendaciones de Hardware.

Servidor Soportadas	laptops	procesador	RAM	Almacenamiento
Pequeño	<20-25	466 Mhz	1 GB	40-120 GB
Grande	<150	1 Ghz	1 GB	320-750 GB

Primera etapa: Se inicia al ubicar una escuela piloto donde se pueda implementar un servidor escuela, luego se debe identificar el numero de laptops XO que ha sido desplegada en esta escuela y si cuentan con acceso a internet. De acuerdo al número de laptops XO se identifica el escenario. A continuación se describen los diferentes tipos de escenarios:

- **Escenario de escuela pequeña:** Se debe implementar un (1) servidor escuela.

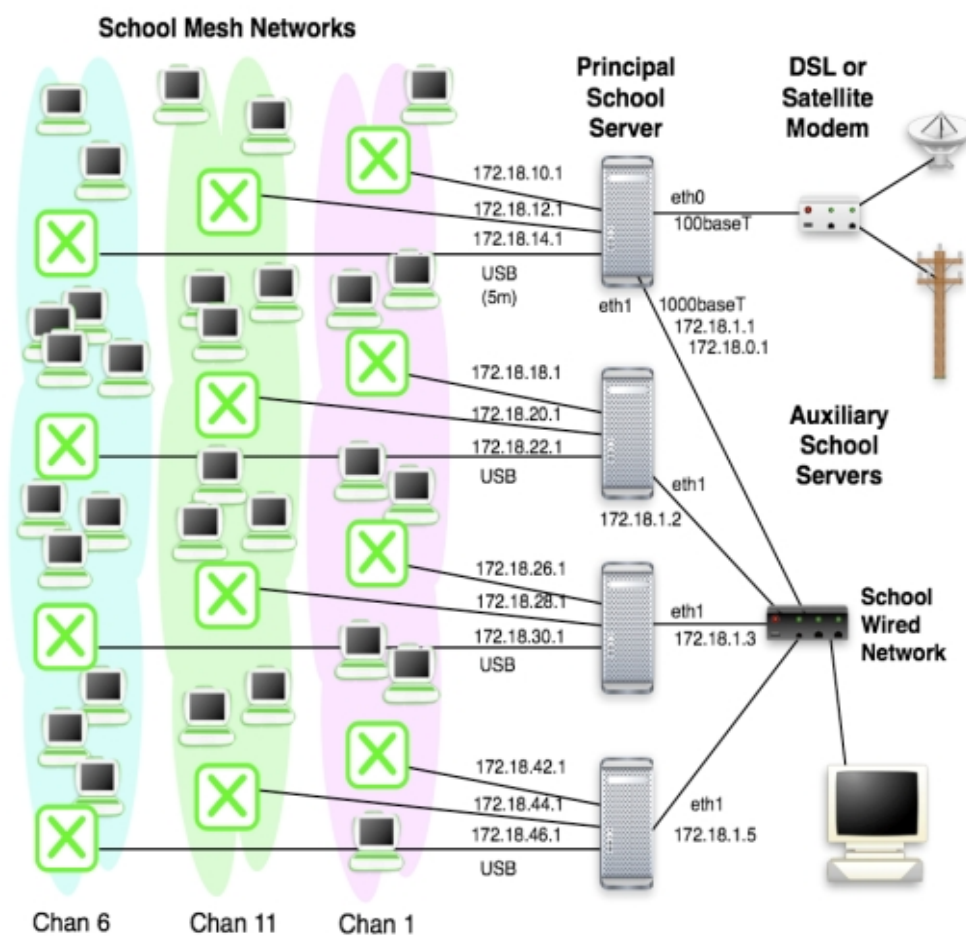
Figura N° 2: Escenario de escuela pequeña



Fuente: wiki.laptop.org

- **Escenario de escuela grande:** Se implementa un servidor escuela que permita que se pueda incrementar servidor XS auxiliares dependiendo del número de laptops XO) según la capacidad del servidor.

Figura N° 3: Escenario de escuela grande



Fuente: wiki.laptop.org

En el proyecto se tomará el Escenario de escuela pequeña que cuenta con un servidor Escuela XS (principal) en una escuela piloto.

En esta etapa se debe tener en cuenta los siguientes recursos:

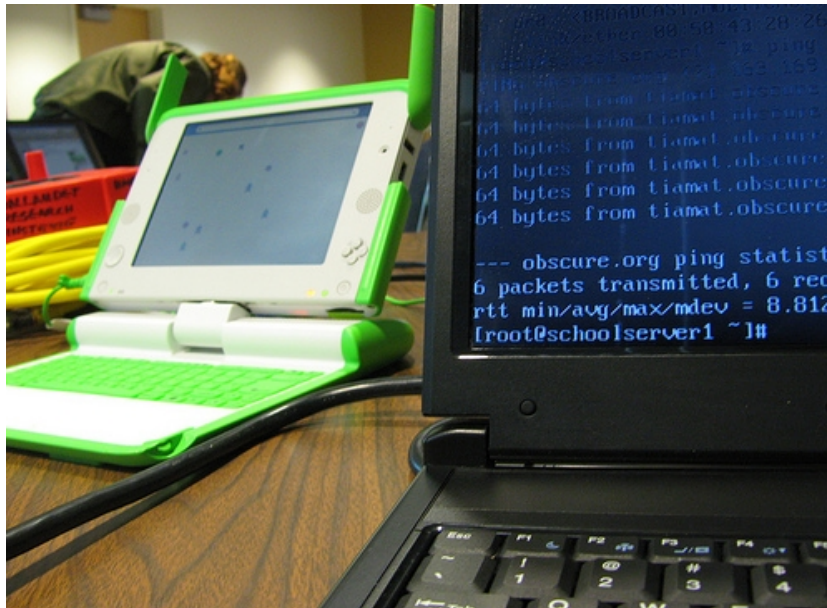
1. Un Access point.
2. Un Switch (si la escuela cuenta con Computadoras adicionales).
3. Una PC (servidor XS).
4. Cable UTP
5. Conectores RJ45

Se debe identificar la ubicación de la escuela, acondicionar el espacio físico donde deba ir el servidor XS y a cuánta distancia se encuentra de las aulas con las laptops XO.

Segunda etapa:

En esta etapa se debe configurar los servicios necesarios de la Escuela para que las laptops XO se puedan comunicar con el servidor Escuela y puedan acceder a servicios tales como: proxy (restricciones de páginas), jabber (comunicación), actualizaciones de software como la interface Sugar y las actividades que vienen a ser los programas educativos y servicios adicionales que la escuela requiera para su óptimo funcionamiento e interacción.

Figura N° 4: Laptop XO conectada al servidor Escuela

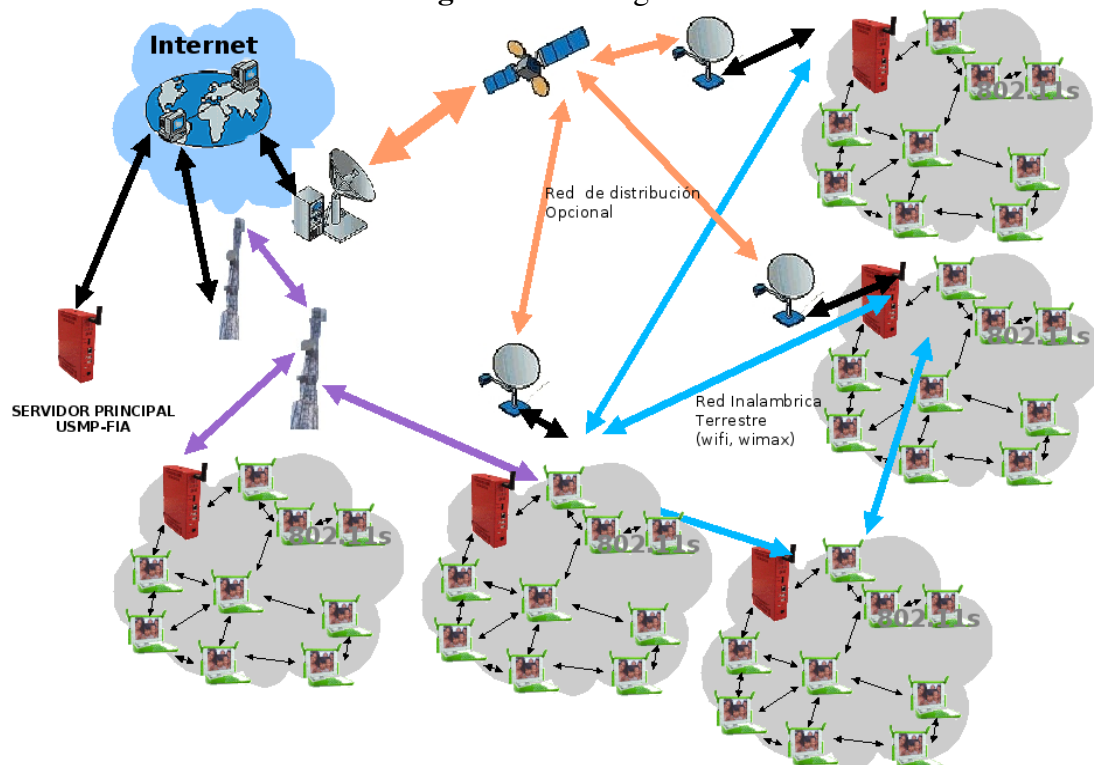


Fuente: www.olpcnews.com

Tercera etapa:

Implementar un servidor principal en la FIA USMP que interactue con el Servidor escuela implementado en la Segunda Fase, donde se almacenen los registros de las laptops XO conectadas al servidor XS de la escuela piloto, las notificaciones de fallas de laptops, actualizaciones de software y consultas por parte de los docentes, como resultado se mostrará reportes que faciliten la identificación de problemas en las escuelas y brindar una solución adecuada.

Figura N° 5: Diagrama



3. CONCLUSIONES

El siguiente proyecto propuesto, permite tener una mayor interacción y comunicación con las escuelas de zonas rurales identificando los posibles problemas, como de hardware, software y dar un solución viable, los reportes de las escuelas nos permite tener detallado el progreso del proyecto OLPC implementado y un mayor control.

4. REFERENCIAS

- [1]OLPC “One Laptop Per Child” <http://wiki.laptop.org>
- [2] OLPC “One Laptop Per Child” http://wiki.laptop.org/go/School_server
- [3] OLPC “One Laptop Per Child”
[http://wiki.laptop.org/go/XS Configuration Management](http://wiki.laptop.org/go/XS_Configuration_Management)
- [4]Ministerio de Educación del Perú
http://www.perueduca.edu.pe/olpc/OLPC_Home.html
- [5]OLPC NEWS <http://www.olpcnews.com>

Modelación Formal Espacio-Temporal: Uso de Cálculo de Eventos Basado en Objetos

Cristian Vidal Silvaⁱ, Elizabeth Vidal-Duarteⁱⁱ,
Ana Zuñiga Baezaⁱⁱ, Lorena Albornoz Pobleteⁱⁱⁱ

ⁱ Ingeniería Informática Empresarial, Universidad de Talca, Talca – Chile
cvidal@utalca.cl

ⁱⁱ Ingeniería de Sistemas, Universidad Católica San Pablo, Arequipa – Perú
e.vidal@usp.edu.pe

ⁱⁱⁱ Ingeniería Informática Empresarial, Universidad de Talca, Talca – Chile
{azuniga, lalbornoz}@alumnos.utalca.cl

Abstract. La modelación formal de sistemas software, es una labor no tradicional en la implementación de aplicaciones software. La modelación formal de sistemas de alto costo (sistemas críticos), es una labor que con ligera relevancia y/o vigencia en los últimos años. Un sistema de información espacio-temporal es un ejemplo de sistema crítico, debido al alto costo de la información manipulada por él. Este trabajo detalla extensiones realizadas al formalismo lógico Cálculo de Eventos basado en Objetos (OEC), para la modelación e implementación de sistemas espacio-temporales.

Keywords: OEC, Modelación, Espacio-Temporal, Objetos, Eventos.

1 Introducción

La modelación formal y especificación formal de requerimientos de sistemas espacio-temporales o sistemas temporales geoespaciales, es una labor no tradicional en ingeniería de software, ciencias de la computación y geoinformática. Este trabajo presenta un marco formal con conceptos de orientación a objeto, para la modelación de entidades espacio-temporales. La principal contribución de este trabajo, es la propuesta de un marco lógico con conceptos de orientación a objeto, de objetos y eventos espacio-temporales, para la definición de un modelo conceptual de una base de datos espacio-temporal y su correspondiente implementación como una base de datos deductiva. Cabe señalar, que la especificación formal de entidades espacio-temporales es determinante en la definición de sistemas de bases de datos espacio-

temporales. El marco formal propuesto, es la extensión de Object Event Calculus (OEC).

Este artículo, es un resumen de los principales resultados obtenidos en la Tesis de Magíster en Ciencias de la Computación de uno de los autores (Vidal, 2007).

El marco lógico, expuesto en este trabajo, provee mecanismos de razonamiento para derivar el estado de un objeto en el tiempo (el valor de los atributos de un objeto en el tiempo), para derivar la membresía en una clase de un objeto en el tiempo, para establecer el período de tiempo de la ocurrencia de un evento, y para determinar las relaciones que varían con el tiempo entre objetos, eventos o entre objetos y eventos, además de dar un soporte a las clases de eventos principales en el entorno espacio-temporal, y presentar una asociación de agregación entre objetos variante en el tiempo.

2 Modelación Espacio-Temporal

2.1 Modelación Snapshots

Uno de los primeros enfoques de modelación de información espacio-temporal orientado al estado de los objetos lo constituye el modelo basado en snapshots de objetos. Este modelo fue propuesto inicialmente en (Armstrong, 1988) y permite la representación de una colección de unidades temporalmente homogéneas de un tema en particular. De una manera más simple, los snapshots son modelos o imágenes del estado del mundo en instantes particulares de tiempo (fotografías de los objetos en estudio). En la *Figura 1* (Abdelmoty, 1994), se puede apreciar una ilustración del modelo de snapshots. En ciertos instantes de tiempo T_i , se construye una región o conjunto de snapshots, donde cada celda o subregión de cada una de las regiones, corresponde a un snapshot (fotografía) de un tema particular. De esta manera, es posible analizar modelo de snapshots como el muestreo de un fenómeno dinámico en una secuencia de instantes temporales (Abdelmoty, 1994), idea propia de Procesamiento de Señales (Vidal, 2007). Esta visión del modelo de snapshots introduce una serie de problemas prácticos, los que se analizarán a continuación.

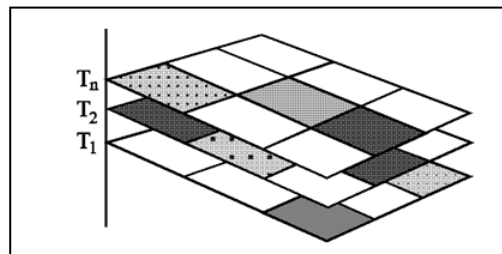


Figura 1. Un ejemplo del modelo de snapshots.

De una manera más simple, el modelo de snapshots no puede responder con certeza a consultas que involucren intervalos de tiempo entre snapshots sucesivos: una representación basada en snapshots sólo puede extraer cambios en un objeto comparando diferentes snapshots (fotografías) del objeto en cuestión y no permite saber el instante exacto en el tiempo cuando ocurre un cambio sobre un objeto (Worboys, 1994).

2.2 Modelación Estados

El modelo basado en estados introduce la noción de marca de tiempo (timestamp) al conjunto de objetos en estudio. En un modelo basado en estados de objetos, la idea es tener un registro de todos los estados temporales de un objeto. El estado de un objeto está completamente definido según el valor de sus atributos. Así, cuando un objeto cambia el valor de cualquiera de sus atributos, entonces dicho objeto experimenta un cambio de estado.

Un modelo muy utilizado en implementaciones prácticas de bases de datos temporales emplea un marca de tiempo basado en intervalos de tiempo, tanto para el tiempo de transacción como para el tiempo válido (T_s - T_e y V_s - V_e , respectivamente) (Chen, 1989). Estas marcas de tiempo se pueden asociar a cualquier relación dinámica en la que participan los objetos. Por ejemplo, si se piensa en una persona en particular como 'Cristian', y los empleos en los cuales esta persona ha trabajado, entonces esta relación dinámica (*Trabaja_En*) podría tener asociada una marca de tiempo. En la *Tabla 1*, se pueden apreciar el ejemplo de una instancia de la relación dinámica *Trabaja_En*, donde se emplea un intervalo de tiempo para representar el tiempo válido de cada una de las tuplas en la relación. Cabe señalar que la granularidad temporal para todas las tuplas en la relación es diaria.

TABLA 1. EJEMPLO DE RELACIÓN TRABAJA_EN EN UN MODELO BASADO EN ESTADOS.

Persona	Empleo	Lugar	V_s - V_e
Cristian	Programador	Serviu	01 de Enero 2000 - 01 de Marzo de 2000
Cristian	Programador	Serviu	01 de Enero 2001 - 07 de Marzo 2001
Cristian	Ingeniero	Cementos Bio Bio.	01 de Julio 2005 - NOW

En el ejemplo anterior se muestra la utilización del valor **NOW** para un extremo del intervalo de tiempo, el cual permite representar un intervalo de validez abierto en su extremo derecho (o de validez actual) (Gregersen, 1999). Al momento de responder a una consulta que incluya atributos con valor **NOW**, este valor debiera reemplazarse por el valor del tiempo actual.

En general, el modelo basado en estados permite responder a una amplia gama de consultas espacio-temporales. No obstante, en su concepción, el modelo deja fuera un conjunto importante de información referente a los fenómenos que ocasionan los cambios de estado de un objeto. Dicho modelo sólo se centra en los estados de los

objetos, pero no dice nada acerca de los eventos que afectan el estado de los objetos en estudio.

2.3 Modelación Eventos

Un modelo basado únicamente en eventos permite el registro de la ocurrencia de eventos espacio-temporales, y a partir de dichas ocurrencias, deriva el estado temporal de cualquier objeto. A nivel conceptual, un modelo basado en eventos modela las relaciones de *Participación e Involucramiento* entre objetos y eventos (Grenin, 2004) (Grenon, 2002). A nivel lógico y físico, un modelo basado en eventos deriva el estado de los objetos, al estilo del Cálculo de Eventos basado en Objetos (Chen, 1989) (Kesim, 1996) (Vidal, 2005) (OEC).

Un modelo basado en eventos es un modelo completo, ya que registra todas las ocurrencias de eventos que afectan a los objetos del modelo. En un modelo de este tipo se restringe, usualmente, el conjunto de eventos en estudio según la información que se desea destacar en el dominio del problema. De esta manera, se evita la carga de información innecesaria e irrelevante, ya que la derivación del estado de un objeto puede, por lo general, ser una tarea con un alto costo computacional.

Si bien este modelo permite responder a cabalidad a todas las consultas espacio-temporales, se ha constatado de manera empírica el alto costo computacional que acarrea la utilización de este modelo en la derivación de estados de objetos espacio-temporales. Esto radica principalmente en que no existe un estado base de cada uno de los objetos en cuestión, lo que obliga a la revisión de un alto número de tuplas para poder dar respuesta a consultas espacio-temporales básicas.

3 Especificación Basada en Lógica para Objetos y Eventos

3.1 Lógica para Objetos Complejos

Existen muchos intentos que usan la lógica para establecer una semántica formal de objetos complejos (Armstrong, 1988). Gran parte de estos trabajos han centrado la mayor parte del esfuerzo en el desarrollo de nuevas lógicas para dar soporte a los conceptos de orientación a objeto. Todas estas lógicas deberían manejar la noción de identidad de un objeto, objetos complejos, clasificación de objetos, y la propiedad de herencia.

En este trabajo se utiliza C-Logic (Chen, 1989) para la especificación de un modelo de datos que incluye objetos y eventos. La principal ventaja de utilizar C-Logic es el mapeo directo de las especificaciones escritas en este lenguaje a fórmulas lógicas de primer orden. Se utilizan predicados unarios para tipos y predicados binarios para atributos. C-Logic permite que la información de una entidad se especifique en partes o piezas acumulativas de información, lo cual facilita la modificación de subpartes de la especificación de un objeto. Este tipo de

especificación puede ser fácilmente implementada en lenguajes de programación lógica, como por ejemplo Prolog.

En C-Logic, una clase de objetos *Objetc_Class*, con las propiedades o atributos *attribute₁, ..., attribute_n*, es especificada como:

$$class : Object_Class[attribute_of \Rightarrow \{attribute_1, \dots, attribute_n\}]$$

Usando C-Logic, los objetos complejos son considerados como colecciones o conjunciones de propiedades atómicas. Cada atributo es visto como un predicado binario y cada símbolo de clase como un predicado unario. Un objeto con varias etiquetas de atributo puede ser descrito como una conjunción de varias fórmulas atómicas en lógica proposicional.

3.2 Cálculo de Eventos Basado en Objetos

Una propuesta, denominada Cálculo de Eventos Basado en Objetos (OEC), modifica el Cálculo de Eventos Original (Cálculo de Eventos Relacional) (Kesim, 1996) para describir y razonar acerca de los cambios que afectan a los atributos de un objeto en un marco lógico (Worboys, 1994). En OEC, un eventos *Ev* inician o terminan valores *V* de atributos *Attr* de un objeto *Obj* a través de las siguientes reglas:

$$initiates(Ev, Obj, Attr, V).$$

$$terminates(Ev, Obj, Attr, V).$$

De manera similar al Cálculo de Eventos clásico, en el OEC la ocurrencia de eventos es especificada con un predicado que relaciona el evento *Ev* con el tiempo *T*, donde *T* es el tiempo válido de la ocurrencia de un evento.

$$happens(Ev, T).$$

Con este formalismo, es posible derivar el valor de un atributo en un instante de tiempo dado, por medio de las siguientes reglas:

$$\begin{aligned} holds_at(Obj, Attr, V, T) \leftarrow \\ & happens(Ev, T_s), \\ & T_s \leq T, \\ & initiates(Ev, Obj, Attr, V), \\ & \neg broken(Obj, Attr, V, T_s, T). \end{aligned}$$

$$\begin{aligned} broken(Obj, Attr, V, T_s, T_e) \leftarrow \\ & happens(Ev, T), \\ & T_s \leq T \leq T_e, \\ & terminates(Ev, Obj, Attr, V). \end{aligned}$$

Estas reglas indican que el atributo *Attr* del objeto *Obj* tiene un valor *V* en el tiempo *T* si ocurre un evento *Ev* en el tiempo T_s (*happens*(*Ev*, T_s)), donde $T \geq T_s$, el evento *Ev* inicia el atributo *Attr* del objeto *Obj* con el valor *V*, y este valor *V* en el atributo *Attr* no ha sido terminado en el intervalo de tiempo $[T_s, T]$.

La derivación del valor de un atributo de un objeto puede ser también definida para un intervalo de tiempo cerrado o abierto mediante las siguientes reglas: *hods_for*, *holds_since*, *terminated*, *broken_aux*, y *terminated_after*, todas ellas descritas en (Vidal, 2007).

4 Relación de Agregación de Objetos en OEC

La agregación es una abstracción fundamental de la orientación a objetos (González, 1996). Esta relación establece que un conjunto de objetos forma una composición u objeto compuesto de más alto nivel (Hornsby, 2000); es decir, existe una clase cuyas instancias son objetos compuestos y representan una forma de *todo*, y otra clase cuyas instancias (objetos parte) son *parte* de alguno de estos todos

La relación de agregación es de gran importancia en el contexto de modelación de datos espaciales (Hornsby, 2000) (Vidal, 2004). La agregación tiene dos versiones: una versión débil (denominada simplemente agregación) y una versión fuerte denominada composición. En agregación débil, el tiempo de vida del objeto parte no está determinado por la existencia del todo. En agregación fuerte, el tiempo de vida de las partes está directamente determinado por la existencia del todo; esto es, cuando el todo deja de existir, las partes también lo hacen. Un objeto agregado recibe el nombre de objeto compuesto y es una forma de objeto complejo. Las diferentes formas de objeto compuesto han sido estudiadas en profundidad, pero de manera informal, en el contexto de SIG (sistemas de información geográfica) (Hornsby, 2000) (Vidal, 2004). Estos objetos juegan un rol importante en la representación de conocimiento espacio-temporal y en la formulación de consultas espacio-temporales (Vidal, 2004). La Figura 2 (Vidal, 2004), es un ejemplo de composición en UML. En esta relación, se aprecia que una “region” puede estar compuesta *débilmente* de una o más instancias de la clase “provincia”, y una instancia de la clase “provincia” puede estar compuesta *fuertemente* de una o muchas instancias de la clase “comuna”. Esta modelación, refleja de manera simple el proceso de regionalización Chilena, ya que se da el hecho de que ninguna provincia ha sido eliminada, y si se ha destruido o eliminado a alguna de las regiones, sin la eliminación de las provincias asociadas. Este ejemplo modela, de manera simplificada, la dinámica del proceso de regionalización de Chile, desde 1950 a la fecha.

Para especificar (indicar) que un objeto es parte de otro, se utiliza un predicado particular *is_part*. A diferencia de la relación de herencia *is_a* entre clases, la relación de agregación *is_part* define una relación semántica entre clases y entre objetos. La relación de agregación entre clases define una relación semántica que afecta a todas las instancias de las clases; sin embargo, por lo general, las consultas siempre se refieren a la relación de agregación entre instancias particulares de las clases. Por esta razón, la formalización aquí presente, se refiere a la agregación y

composición entre objetos. Una característica importante de la relación *is_part* es que ésta puede ser dinámica o variable en el tiempo.

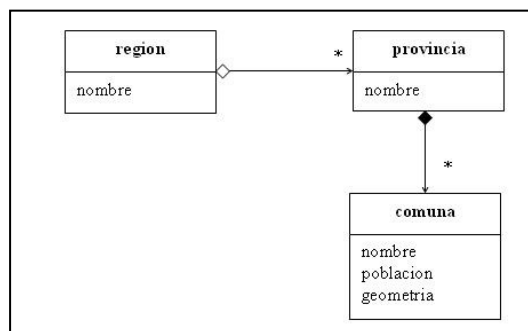


Figura 2: Ejemplo de Agregación y Composición de Clases UML.

5 Clases de Eventos Espacio-Temporales

Para la representación de eventos en un modelo espacio-temporal orientado a objetos, se deben definir clases de eventos para la definición de instancias de eventos que en su ocurrencia trabajen sobre objetos del modelo. En este trabajo, cada una de las clases de eventos definidas, es una subclase o hereda la naturaleza espacial de alguna de las clases de eventos espaciales presentes en el modelo. Las clases de naturaleza espacial permiten la definición de eventos o instancias (objetos) para la creación de objetos de alguna clase, destrucción de objetos de alguna clase, cambio de objetos de alguna clase, división de objetos de alguna clase y fusión de objetos de alguna clase. Al momento de definir una clase de eventos, se indica la naturaleza del evento mediante una relación de herencia *is_a*, y se establece la clase de objetos a los cuales afectarán las instancias de dicha clase de eventos y, opcionalmente, los atributos de cada una de las clases de eventos. La especificación de clases de eventos, usando C-Logic, se muestra a continuación.

$$\begin{aligned}
 &event_class : Ev_Class[is_a \Rightarrow Sup_Class_Ev, \\
 &\{attribute_1 \Rightarrow Attr_1, \dots, attribute_n \Rightarrow Attr_n\}, \\
 &involvement \Rightarrow Obj_Class].
 \end{aligned}$$

Los predicados en Prolog, equivalentes a esta regla C-Logic, son los siguientes:

```

event(Ev_Class).
attribute(Ev_Class,Attr1).
....
is_a(Ev_Class,Sup_Class_Ev).
involvement(Ev_Class,Obj_Class).
  
```

En estas reglas se define una clase de eventos *Ev_Class*, con un conjunto de posibles atributos ($Attr_1, \dots, Attr_n$). Se define la naturaleza espacial de una clase a través del predicado *is_a*, que señala una relación de herencia entre la clase definida (*Ev_Class*) y una clase de naturaleza espacial que se trabaje en el modelo. Centrándose en el dominio espacial, se describen 5 diferentes tipos o clases de eventos y sus relaciones con objetos. Los 5 tipos de eventos de naturaleza espacial considerados en este trabajo son *Create*, *Change*, *Destroy*, *Split* y *Merge*. Para establecer con que instancia(s) de clase de objetos interactuará cada instancia u objeto de cada clase de eventos, se usa el predicado *involvement*.

En la definición de instancias de eventos, a diferencia de los atributos de un objeto, los atributos de un evento se asignan con el predicado del evento mismo. Mientras los atributos de un objeto son derivados a través de predicados en términos de eventos, los valores de los atributos de eventos son determinados de la especificación de la ocurrencia del evento. En este sentido, los valores de los atributos de un evento no cambian con el tiempo, aún cuando los eventos sean temporales por naturaleza. En este modelo, en el predicado de evento, se incluye un atributo específico *class_event*, el cual indica la clase a la cual pertenece el evento.

$$\begin{aligned}
 &event : Ev [instance_event \Rightarrow Class, \\
 &\left. \begin{array}{l} values_event_1 \Rightarrow (Attr_1, Val_1), \dots, \\ values_event_n \Rightarrow (Attr_n, Val_n) \end{array} \right\} \\
 &\left. \begin{array}{l} object \Rightarrow Obj_1, values_object_1 \Rightarrow \\ (Obj_1, Attr_Obj_1, Val_Attr_Obj_1), \dots, \\ object \Rightarrow Obj_m, values_object_m \Rightarrow \\ (Obj_m, Attr_Obj_m, Val_Attr_Obj_m) \end{array} \right\}]
 \end{aligned}$$

A continuación se describen la clase de eventos espacio-temporales para la creación de objetos. En (Vidal, 2007) se detalla las otras clases de eventos espacio-temporales.

5.1 Eventos de Creación (Create)

Para la definición de instancias de la clase de eventos de creación de objetos (*Ev*), se asignan valores a los atributos del evento y se establece él o los objetos que serán afectados directamente por la ocurrencia de dicho evento.

$$\begin{aligned}
 &instance_event(Ev, Class_Ev). \\
 &values_event(Ev, Class_Ev_Attr_1, Val_Ev_Attr_1). \\
 &\dots \\
 &values_event(Ev, Class_Ev_Attr_n, Val_Ev_Attr_n).
 \end{aligned}$$

$$\begin{aligned}
&object(Ev, Obj_1). \\
&values_object(Ev, Obj_1, Class_Obj_Attr_1, Val_Attr_Obj_1). \\
&\dots \\
&object(Ev, Obj_m). \\
&values_object(Ev, Obj_m, Class_Obj_Attr_1, Val_Attr_Obj_1).
\end{aligned}$$

Un evento de creación de objeto espacio-temporal es aquel evento que crea o inicia un objeto con algún atributo espacial.

Un evento de creación de objetos, asigna los objetos creados como parte de una clase, la cual es especificada a través del predicado *involvement* en la definición de la clase del evento.

$$\begin{aligned}
&initiates(Ev[instance_event \Rightarrow Class_Event], Obj, \\
&Attr, Val) \leftarrow is_a(Ev_class, creation), \\
&object(Ev, Obj), \\
&values_object(Ev, Obj, Attr, Val). \\
\\
&assigns(Ev[instance_event \Rightarrow Ev_class], Obj, Class) \leftarrow \\
&Ev[instance_event \Rightarrow Ev_class], \\
&is_a(Ev_class, creation), \\
&involvement(Ev_class, Class), \\
&object(Ev, Obj, new).
\end{aligned}$$

En estas reglas, el evento *Ev* es una instancia de la clase de eventos *Ev_class*, que hereda propiedades de la clase de eventos *creation* (*is_a(Ev_Class, creation)*), la que permite la creación de objetos de una clase y asignar valores a los atributos del objeto creado. La primera regla asigna el objeto *Obj* a la clase *Class* y la segunda regla inicia el objeto *Obj* con el valor *Val* en el atributo *Attr*.

6 Conclusiones

En este trabajo se ha abordado el tema del uso de formalismos para la modelación formal de un sistema espacio-temporal basada en los conceptos de objeto y evento. Se han presentado los formalismos y enfoques de modelación clásicos usados para la labor de modelación e implementación de una base de datos temporal y/o espacio-temporal. Se ha presentado además, un formalismo relativamente actual, que permite una implementación de una base de datos temporal o espacio-temporal en ambiente deductivo. Este formalismo es *Cálculo de Eventos Basado en Objetos* u OEC. Se puede concluir que el uso de este formalismo permite la implementación de una base de datos deductiva o sistema deductivo sin mayor dificultad, gracias a su conjunto de reglas. El uso de OEC, en una base de datos no deductiva, requiere características no

presentes en las bases de datos normales, como lo es el uso de enfoques recurrentes ampliamente usado en los enfoques deductivos.

En orientación a objeto, se ha mostrado una formalización en OEC de la relación de agregación de objetos, detallada en (Vidal, 2007) (Vidal, 2004), y las modificaciones necesarias a las reglas del OEC para la derivación de estados y membresía temporal de objetos, y para la formalización de clases de eventos de naturaleza espacial, y establecer relaciones entre dichas clases. Cabe señalar que aún resta por analizar en profundidad, el overhead en el sistema por las reglas no presentes en OEC original, con una mayor cantidad de información. Es posible extender el EOC, para el manejo de cambio continuo, tal y como fue hecho en el calculo de eventos original en (Chomicki, 1998).

Referencias

1. Vidal, C. (2007). Un Marco Formal para Base de Datos Espacio-Temporal Basado en Objetos y Eventos. Tesis de Magister en Ciencias de la Computación, Universidad de Concepción.
2. Armstrong, M.P. (1988). Temporality in spatial databases. *Proceedings: GIS/LIS'88*, 2:880-889.
3. Abdelmoty, A. (1994). Geographic data handling in a deductive object-oriented database. In D.Karagiannis, 23editor, *International Conference on Database and Expert Systems Applications DEXA'94*, 445-454. Springer-Verlag.
4. Worboys, M. (1994). A Unified Model of Spatial and Temporal Information. *Computer Journal*, 37(1):26-34.
5. Chen, W. (1989). C-logic of Complex Objects. *ACM SIGACT-SIGMOD-SIGART Symp. Principles of Database Systems*, pages 369-378.
6. Gregersen, H. (1999). Temporal Entity-Relationship Models—A Survey. *IEEE Transactions on Knowledge and Data Engineering*, 11(3):464-497, 1999.
7. Grenin, P. (2004). Snap and span: Towards Dynamic Spatial Ontology. *Journal of Spatial Cognition and Computation*, 4(1):69-103.
8. Grenon, P. (2002). The Formal Ontology of Spatio-Temporal reality and its formalization. AAAI.
9. Kesim, F.N. (1996). A Logic Programming Framework for Modeling Temporal Objects. *IEEE Transactions on Knowledge and Data Engineering*, 8(5):724-741.
10. Vidal, C. (2005). A Logical Approach for Modeling Spatio-Temporal Objects and Events. *CoMoGIS'05, LNCS*.
11. González, C. (1996). *Sistemas de Bases de Datos*. Editorial Tecnológica de Costa Rica, Primera Edición.
12. Hornsby, K. (2000). Identity-Based Change: A Foundation for Spatio-Temporal Knowledge Representation. *International Journal of Geographic Information Science*, 14(3):207-244.
13. Vidal, C. (2004). Una Formalización de la Relación de Agregación de Objetos usando Cálculo de Eventos Orientado a Objeto. XIII Encuentro Chileno de Computación, Valdivia Chile.
14. Chomicki, J. (1998). *Logics for Databases and Information Systems*. Kluwer Academic Publishers.

Planificación y seguimiento de la trayectoria de un robot móvil dentro de un ambiente controlado

Liz Jave¹²

Smith Arauco¹²

Keitel Cervantes¹²

Jacinto Colán¹²

¹ Centro de Tecnologías de la Información - CTIC

² Universidad Nacional de Ingeniería - UNI

liz.ahs@gmail.com sac2412@hotmail.com kacsss@gmail.com colan1804@gmail.com

Resumen

Planificar y hacer el seguimiento de la trayectoria de un robot móvil (RM), mediante técnicas de Visión Computacional (VC), en un ambiente con obstáculos inmóviles (ambiente controlado AC).

1. Introducción

Para realizar determinadas tareas los RM necesitan de seguridad en su traslado, evitando los obstáculos que puedan encontrar en el camino. Si el ambiente donde se va a desplazar el RM es totalmente conocido se generan trayectorias libres a partir del mapa que se tiene del ambiente. Los métodos para determinar la trayectoria no son confiables cuando el ambiente varía, la incertidumbre del mapeo crece o la información acerca del ambiente es insuficiente. Es por ello que se necesita un planificador de trayectorias confiable que permita determinar la posición del RM y decidir su siguiente posición.

Un planificador de trayectorias se puede definir como el conjunto de técnicas que permiten a un RM llegar a su objetivo teniendo en cuenta el ambiente que lo rodea. Es aquí donde contamos con dos problemas el de planificar la ruta y el de su seguimiento.

Estos dos subproblemas son enfocados desde la VC. El método de Dubin y el uso de splines darán una trayectoria inicial, luego se hace el seguimiento de la trayectoria determinando la orientación del RM en cada instante mediante VC.

2. Trabajos Previos

El proyecto se basa principalmente en el trabajo realizado por Ponce desarrollado en el Centro de Inteligencia Artificial del Instituto Tecnológico y de Estudios Superiores de Monterrey, España. Ponce propone una serie de métodos destinados a la obtención de la trayectoria mínima a realizar por un vehículo para llegar a su destino en un área reconocida por una cámara. Se basa en el algoritmo NF1 y no toma consideraciones holonómicas para reducir el problema. También hace referencia al procesamiento de imágenes necesario para el seguimiento del robot y verificación de su trayectoria.

El objetivo del proyecto es la implementación y optimización de sus procedimientos y probar la eficacia de los mismos. Para ello se utilizó las librerías de OpenCV¹, desarrolladas en lenguaje C++ para el tratamiento de imágenes.

¹La librería se puede obtener en <http://sourceforge.net/projects/opencvlibrary>

3. Metodología

El proyecto involucra el desarrollo de 3 áreas: La adquisición y preprocesamiento de la imagen, la planeación de la trayectoria y por último el seguimiento de la misma por parte del RM a través de un sistema de VC.

3.1. Adquisición y Preprocesamiento de la Imagen

Al tomar una fotografía con una cámara web se pierde una dimensión (al pasar de 3D a 2D) perdiendo así las magnitudes reales de la escena tomada. Es por ello que para calcular la ruta del RM se necesiten parámetros que relacionen la imagen con distancias métricas.

En este trabajo utilizamos el método propuesto por Zhang para obtener los parámetros intrínsecos y extrínsecos de una cámara. Usando el modelo de cámara pin-hole, la representación geométrica del espacio sería como muestra la figura 1:

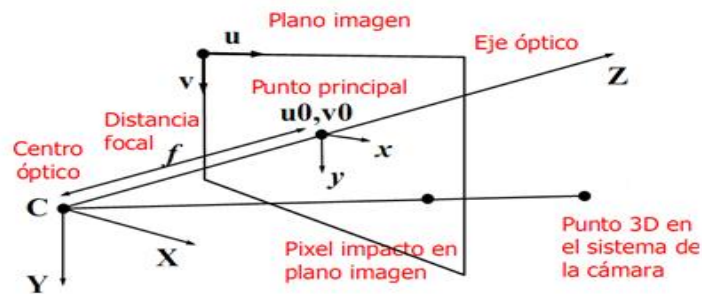


Figura 1: Modelo Geométrico Pin-Hole

3.2. Planeamiento de Trayectoria

Consiste en la obtención de la trayectoria más corta a recorrer para alcanzar un destino señalado. Esta área se inicia con la adquisición de la imagen con vista panorámica del ambiente a trabajar incluyendo al RM, los obstáculos y destino final. Contiene varias sub-etapas presentadas en la figura 2.



Figura 2: Diagrama a bloques de la Planeación de Trayectoria

3.2.1. Reconocimiento y expansión de los obstáculos Se reconocen los obstáculos a través de un procesamiento de imágenes. Dejando de lado las restricciones no holonómicas del RM podemos considerar al RM como un punto expandiendo los obstáculos. En la figura 3 se presenta este proceso.

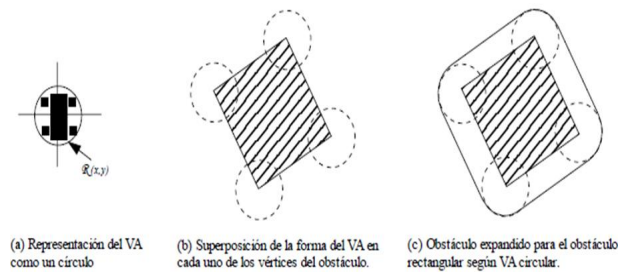


Figura 3: Expansión de obstáculos acorde a las dimensiones del RM

3.2.2. Obtención del C-Space Consiste en la binarización del espacio constituido por los obstáculos expandidos, la posición de inicio del robot y el destino final.

3.2.3. Aplicación del Algoritmo NF1 El método de planificación de trayectorias utilizará una técnica conocida como “Función de Navegación Simple NF1”, la cual se basa en los campos de potencial. El campo de potencial es para zonas que no abarcan la expansión de los obstáculos, para un pixel de la imagen del ambiente se genera en proporción de su distancia al punto meta. Después de aplicar el algoritmo NF1 al Espacio-C la trayectoria se obtiene a partir de un camino generado con un algoritmo de descenso por gradiente. La figura 4 es un ejemplo de la aplicación del Algoritmo NF1.

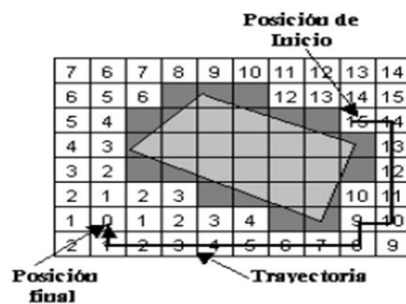


Figura 4: Cálculo de una trayectoria utilizando el Algoritmo NF1

3.2.4. Orientación tangencial con los Caminos de Dubin Después de aplicar el algoritmo NF1, obtendremos una trayectoria compuesta por segmentos de recta que siguen el contorno de los obstáculos. Esta trayectoria es físicamente imposible de realizar por el RM, por ello para tener una trayectoria válida se deben considerar las restricciones no holonómicas en la asignación de puntos de la trayectoria.

La idea es generar trayectorias conocidas como Caminos de Dubin, donde al pasar el RM sobre puntos de control la orientación del RM es tangencial al radio de giro de cada punto de control.

3.2.5. Suavizado por Splines Los caminos de Dubin optimizan la trayectoria en cuanto a distancias que recorre el RM. Sin embargo, son aun difíciles de realizar para el RM puesto que al salir el RM de una curva toma cierto tiempo ajustar su dirección en línea recta, por lo que se tienen cambios bruscos. Por lo tanto, es necesario generar trayectorias que sean continuas y suaves para el RM.

Se hace uso del método de curvas por splines, estas curvas son funciones polinomiales de tercer grado. Estas curvas se definen con base en los puntos de control, realizando el ajuste por interpolación.

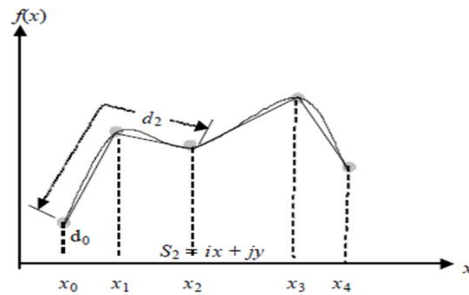


Figura 5: Interpolación por Splines cúbicos

3.2.6. Validación de la Trayectoria Al calcular las curvas de Dubin y suavizar la trayectoria pueden darse casos en los cuales la trayectoria generada se introduzca en alguna región con obstáculos. Para evitar el trazo de la curva sobre esta área, se desplazan los puntos de la trayectoria que se encuentran en colisión hacia el espacio libre.

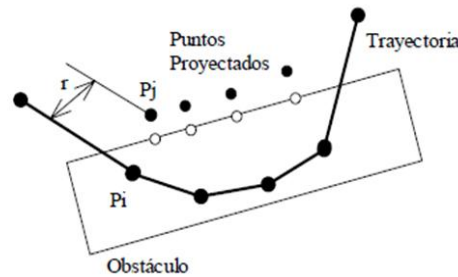


Figura 6: Proyección de los puntos en la zona prohibida

3.3. Seguimiento de la trayectoria

Una vez obtenida la trayectoria a realizar, el problema surge al tratar de asegurar el correcto seguimiento de la misma por parte del robot. Cada cierto tiempo mediante un sistema de VC se obtiene la nueva posición y orientación del RM. Contiene a su vez varias sub-etapas basadas en el procesamiento de imagen realizado por Moreno.



Figura 7: Seguimiento de la Trayectoria

3.3.1. Binarización de la imagen La binarización sirve para identificar regiones en la imagen, permitiendo que el análisis se concentre en la detección de las formas contenidas en la imagen. El objetivo de la binarización es obtener una imagen con sólo dos valores de intensidad, a partir de una imagen en escala de grises o a color; donde un valor de intensidad representa el fondo y el segundo los objetos contenidos en la imagen.

3.3.2. Segmentación de la imagen El propósito de la segmentación es obtener las regiones (segmentos) que forman la imagen, donde las regiones son grupos de píxeles vecinos con la misma intensidad. El algoritmo de segmentación utilizado en este trabajo es un recorrido exhaustivo por toda la imagen colocando una etiqueta a cada región de interés.

3.3.3. Caracterización El objetivo de la caracterización de los objetos, es obtener información que permita la identificación, así como posición y orientación del objeto sobre el espacio de trabajo. Para el reconocimiento del RM en la imagen, se utilizan marcas, que son colocadas en la parte superior del RM. Las marcas son dos círculos de diferente diámetro; por convención, la marca de menor tamaño es el frente del robot. La forma de las marcas es circular, puesto que el centro de la marca se conserva, aún con la deformación por perspectiva.

3.3.4. Reconocimiento y Localización del Robot Cuando se tienen caracterizadas todas las regiones de la imagen, es posible identificar el objeto de interés, como es en este caso el RM. Comparando los valores característicos de cada una de las regiones, encontradas durante la etapa de segmentación y caracterización, con los valores característicos del objeto de interés se reconoce y localiza al RM, solo si estamos dentro del rango de error permitido. Los valores característicos del objeto de interés, son conocidos previamente.

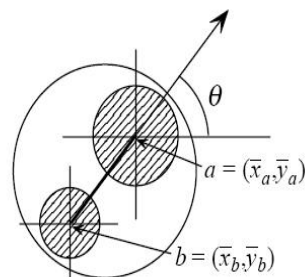


Figura 8: Detección de los Círculos de reconocimiento del robot

4. Implementación y Experimentos

4.1. Implementación

Se adquirió un carro para niños a control remoto. El mando de control remoto se modificó para ser controlado por un microcontrolador PIC 16F877A, el cual se comunica con la PC. Para el control del carrito se empleó el estudio realizado por Hernández para controlar un carrito de juguete.

Para la realización de las pruebas se hizo necesario una interfaz gráfica, figura 9, que sirva de gestor de comunicaciones entre la PC y el mando del carrito.

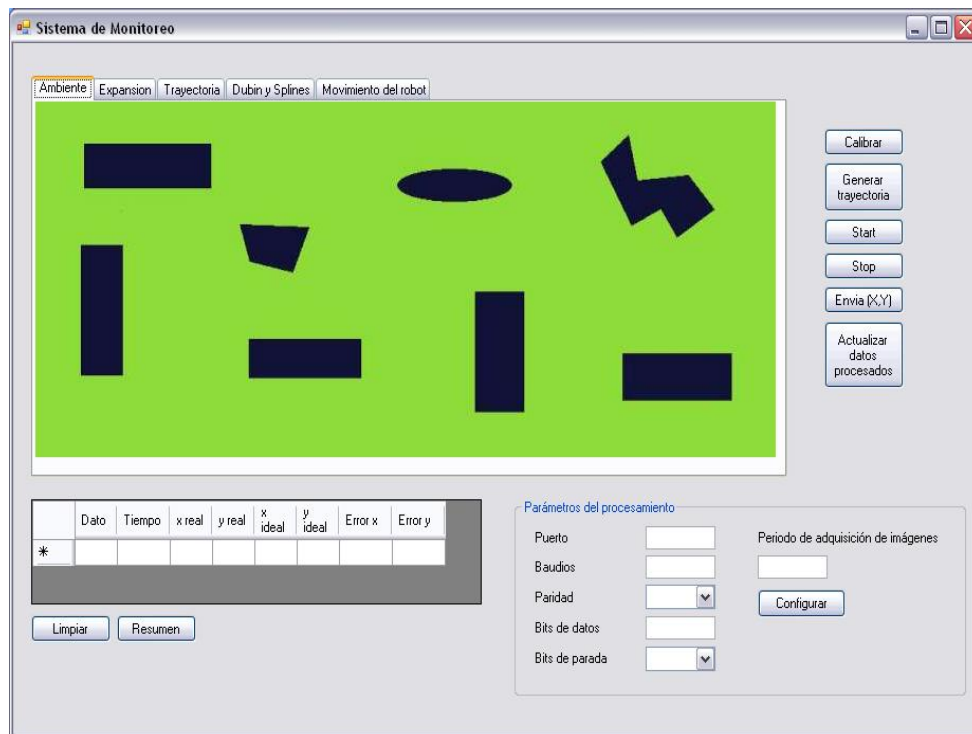


Figura 9: Sistema de monitoreo de la planificación y el seguimiento de la trayectoria del RM

4.2. Pruebas

Se realizaron pruebas en la planeación de la trayectoria con una imagen que simule el ambiente como se muestra en la figura 12.

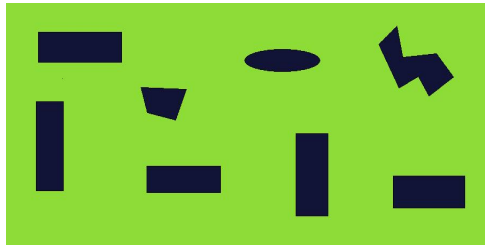


Figura 10: Imagen de prueba del ambiente

La zona de color verde es la posible área de movimiento del RM, las regiones de color negro son los obstáculos que debe de evitar. La imagen 12 muestra el proceso de binarización.

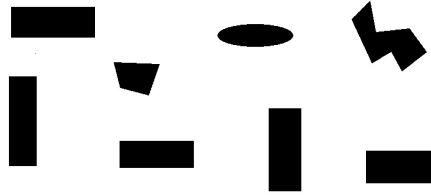


Figura 11: Imagen binarizada

La imagen anterior es la imagen del ambiente que ha sido llevada a solamente dos colores, para poder luego ser segmentada y caracterizada. La imagen 12 muestra el siguiente proceso que es la caracterización.

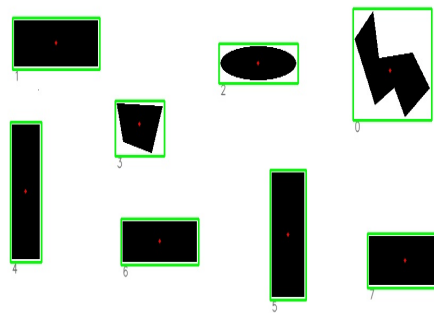


Figura 12: Imagen caracterizada

Para el reconocimiento de la posición del robot se calcularon características de los objetos encontrados. Los resultados se presentan en la tabla 1.

Objeto	Características		
	Area	Perimetro	Compacidad
Obj 1	6601	451,87	2,46
Obj 2	7850	414	1,74
Obj 3	4126	320,51	1,98
Obj 4	3285	252,47	1,54
Obj 5	7592	396	1,64
Obj 6	8235	392	1,48
Obj 7	7850	366	1,74
Obj 8	6116	376	1,57

Cuadro 1: Tabla relacional entre objeto obtenido y sus características

La imagen 14 muestra el algoritmo NF1 para la imagen de prueba.

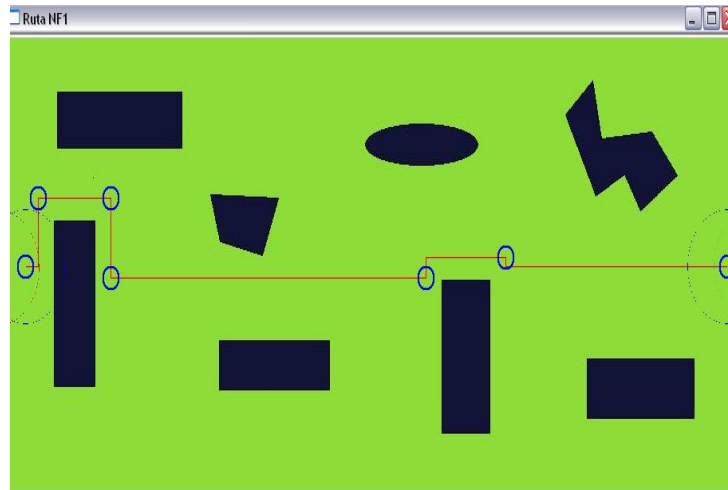


Figura 13: Algoritmo NF1 aplicado a la imagen de prueba

La siguiente imagen muestra la trayectoria discretizada final, aplicandose ya el algoritmo de los caminos de Dubin, suavizado por el metodo de splines y la validación.

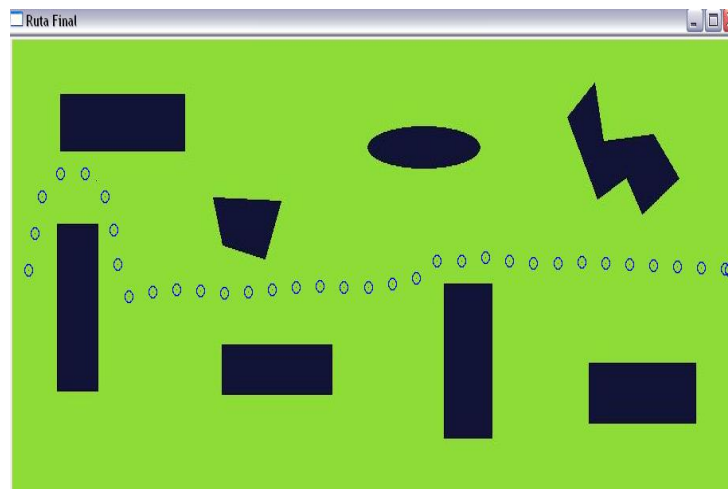


Figura 14: Algoritmo NF1 aplicado a la imagen de prueba

5. Conclusiones

Aún resta realizar pruebas finales relacionadas al control del vehículo y obtención del error en el seguimiento de la trayectoria. Sin embargo el objetivo principal de optimización de los procedimientos se logró. La planeación de trayectorias basadas en el algoritmo de navegación NF1 demostró ser rápido, sencillo y de bajo consumo de recursos computacionales.

El reconocimiento del robot y de su posición se basó en un algoritmo de segmentación y caracterización de objetos que en la mayoría de los casos demostró ser eficaz, preciso y rápido en la localización del robot.

La mayoría de los algoritmos empleados fueron desarrollados por Sedgewick y realizados en Visual Studio 2008, utilizándose la librería OpenCV para el procesamiento de las imágenes, para lo cual se realizó un estudio de su manual de referencia .

Referencias

Cv Reference Manual.

Tutorial de Opencv. http://eupt2.unizar.es/cmedrano/tutorial_opencv.pdf.

Brian Ponce Tavizón (2006). Planeación y Seguimiento Visual de trayectorias para guiar un Vehículo Autónomo.

Jorge Humberto Moreno Scott (2004). Navegación Nodriza de Robots.

Luis Fernando Hernández Camacho (2002). Inclusión de Respuestas Reflejas en el Seguimiento de una Trayectoria por un Vehículo Autónomo.

Robert Sedgewick (1998). Algorithms in C++.

Simulación de Cirugía Plástica

Junior Fabian Arteaga¹ Stivet Fulgencio Obando¹ Jaime Rocca Layza¹

¹ Universidad Nacional de Trujillo

jfabianarteaga@gmail.com, stivet2184@gmail.com, jroccalayza@gmail.com

Resumen

En este trabajo se hace uso de la técnica Warping para obtener deformaciones de una imagen, con el objetivo de simular resultados que se pueden obtener en una cirugía plástica, (Rinoplastia, liposucción, etc.); para este fin mostramos 2 maneras diferentes con las que podemos obtener dichos resultados, la primera es una deformación basada en una transformación a coordenadas polares, y la segunda se basa en el manejo de superficies de Bezier, además de esta técnica se hizo uso de un filtro espacial para suavizar la imagen resultante, este filtro fue definido de manera experimental para nuestra aplicación.

1. Introducción

El procesamiento de imágenes lidia con la manipulación de imágenes, i.e., transformaciones que toman imágenes como entrada produciendo otras imágenes como salida, y todos los problemas que ello pueda generar.

Una de las transformaciones que ha tenido diversas aplicaciones a lo largo del tiempo y que es un importante tópico de reciente investigación es la deformación de imágenes (Warping). Esta técnica ha sido aplicada para corregir las deformaciones introducidas en una imagen. Ejemplos de este tipo de aplicaciones serían: corregir cualquier deformación introducida por las lentes de una cámara, corregir una fotografía aérea la cual viene acompañada de una distorsión asimétrica, que es necesario corregir para propósitos cartográficos. Otro rango de aplicaciones en la que las deformación de imágenes se utiliza ampliamente es en el campo de los efectos visuales cinematográficos (la deformación de imágenes es un paso necesario en el caso de realizar la técnica del morphing o transición suave entre dos imágenes).

En este trabajo se hace uso de la transformada warping para obtener deformaciones de una imagen, con el objetivo de simular resultados que se pueden obtener en una cirugía plástica, ayudando de esta manera a la decisión final de la persona interesada (paciente que desea realizarse una cirugía plástica).

Un warping de imágenes puede ser directamente especificado por una *función de mapeamiento* que establece una correspondencia espacial entre todos los píxeles en la imagen de entrada con la imagen deformada.

En este trabajo se propone dos métodos diferentes para especificar la *función de mapeamiento*: usando coordenadas polares y usando superficies producto tensorial de Bezier.

Este paper se encuentra organizado de la siguiente manera: en la sección 2 explicamos brevemente algunos de los trabajos previos que utilizan esta técnica. En la sección 3, se describe las principales Transformaciones Geométricas. La sección 4 describe la técnica del Warping detallando su fundamento. En la sección 5 se describe la manera como podemos aplicar la transformada Warping para una Simulación de Cirugía Plástica. La sección 6 se describe la transformada Warping para una Simulación de Cirugía Plástica usando la transformación a Coordenadas Polares. Luego en la sección 7 explicamos la técnica del Warping mediante el uso de superficies

producto tensorial de Bezier. En la sección 8 se muestra un Algoritmo en general que resume la manera en la que podemos aplicar esta técnica para obtener una simulación. Los resultados obtenidos son mostrados en la sección 9, en los cuales se muestran comparaciones entre las 2 técnicas presentadas. Finalmente en la sección 10 se hace una discusión de los experimentos obtenidos. Y en la sección 11 se muestran algunas conclusiones obtenidas después de haber experimentado con la técnica presentada.

2. Trabajos Previos

Los trabajos previos realizados, en cuanto a una simulación de cirugía plástica son muy pocos, en el mercado existen algunos software privativos, entre los más conocidos se encuentra VPSS (Virtual Plastic Surgery Software) desarrollado por una compañía francesa (Kaeria SARL), la técnica aplicada por este software se basa en la transformada Warping, más detalles sobre este software pueden ser visto en (Kaeria, 2004).

Otro software que simula el resultado de una cirugía plástica es Face Touch Up, usado por la escuela de medicina de la universidad de Washington y el Instituto de Cirugía de California, más detalles de este trabajo se ve en (Physicians, 2009). Existe también un herramienta del software libre GIMP, el cual realiza deformaciones sobre las imágenes y también puede llegar a simular resultados de cirugía plástica.

3. Transformaciones Geométricas

Las transformaciones geométricas son operaciones que permiten manipular la posición de la imagen en el eje de coordenadas, a si como ajustar la escala y las distintas proyecciones de perspectiva, cuando se aplica un transformación geométrica, es decir el cambio de posición de un pixel a un nuevo lugar, tiende a distorsionarse la calidad visual de la imagen, y es necesario contar con algoritmos para disminuir este problema (algoritmos de interpolación) y la transformada afín en caso de mejorar la escala y la posición de la imagen(Lizardo, 2007). Entre los tipos más importantes de Transformaciones Geométricas tenemos:

- **Escalación.**- Esta transformación aumenta la resolución de la imagen a un tamaño determinado(Lizardo, 2007).

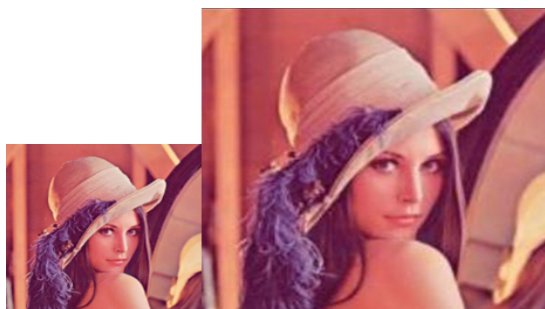


Figura 1: Imagen Original - Imagen a Doble Resolución.

- **Rotación, Traslación.-** La operación de rotación, consiste en girar la imagen un ángulo determinado y la traslación desplaza la imagen una cierta cantidad de píxeles(Lizardo, 2007).



Figura 2: Imagen Rotada - Imagen Traslada.

- **Transformada Afín.-** Esta transformación combina las operaciones de rotación, traslación y escalación (Lizardo, 2007).



Figura 3: Imagen Original - Transformada Afín.

- **Transformada Warping.-** Es una transformación geométrica genérica que modifica la relación espacial de los puntos en la imagen, variando desde una simple traslación o rotación de imagen hasta deformaciones variantes en el espacio bastante complejas(Gomes, 1994).



Figura 4: Imagen Original - Transformada Warping.

4. Warping

La Transformación Warping en general puede realizarse de diversas formas, y la elección de un método en particular dependerá de la aplicación a la que se oriente. Una deformación (Warping) de una imagen puede ser dada directamente por una función (función de mapeo) que proporciona una correspondencia espacial entre todos los puntos de la imagen de entrada, con la imagen deformada (Heckert, 1989).

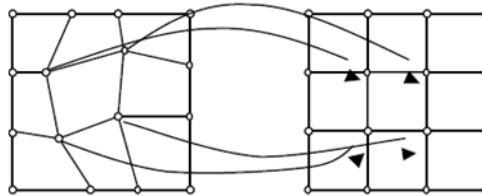


Figura 5: Warping aplicado sobre una malla.

4.1. Fundamento de Warping

Se analiza el fundamento warping a través de dos operaciones: La *especificación* (expresión que describe los resultados deseados por la transformación) y la *implementación* (forma de obtener esos resultados).

Todas las formas de especificación se basan en la idea de definir una transformación para un subconjunto del dominio de la imagen. Este subconjunto, de dimensión 0, 1 o 2, son normalmente especificados en dos situaciones distintas, origen y destino. El origen está asociado con un parámetro del dominio; y el destino, con una deformación de este parámetro. De esta forma, una especificación de Warping puede ser definida como un conjunto de pares ordenados P tal que (Gomes, 1994):

$$P = \{(s_i, d_i) \mid s_i \subseteq U, d_i \subseteq U' \text{ and there exists } W_i : s_i \rightarrow d_i \text{ such that } d_i = W^i(s_i)\} \quad (1)$$

donde $U \subseteq \mathbb{R}^2$ y que en la práctica es un rectángulo alineado con los ejes cartesianos de \mathbb{R}^2 .

La transformación Warping será definida como una transformación $W:U \rightarrow U'$ tal que esta restringido al dominio s_i , $W|_{s_i}$, coincide con una transformación $W_i:s_i \rightarrow d_i$. Esto es equivalente a decir que una transformación W_i es extendida para obtener una deformación Warping W . Es conveniente definir los conjuntos origen (S) y Destino (D) como sigue:

$$S = \{s \subseteq U \mid \exists d \subseteq U' \text{ such that } (s, d) \in P\} \text{ and } D = \{d \subseteq U' \mid \exists s \subseteq U \text{ such that } (s, d) \in P\} \quad (2)$$

Diferentes configuraciones de S e D resultarán en diferentes situaciones de especificación, de las cuales dos son particularmente importantes y, en la práctica, reciben un tratamiento distinto, estas especificaciones son las siguientes: Especificación por Particiones y por Características.

4.1.1. Especificación por Particiones.- En una especificación por particiones, las especificaciones origen y destino S y D definen una partición de los conjuntos U y U' respectivamente. Esto es(Gomes, 1994):

$$(s_0, d_0), (s_1, d_1) \in P \Rightarrow s_0 \cap s_1 = \emptyset \text{ and } d_0 \cap d_1 = \emptyset \quad U_s = \bigcup_{s_i \in S} s_i, \quad U_d = \bigcup_{d_i \in D} d_i \Rightarrow \left\{ \begin{matrix} U_s = U \\ U_d = U' \end{matrix} \right. \quad (3)$$

En la práctica, particiones restrictas tienen elementos de la misma dimensión, o se hace uso de mallas. En la perspectiva basada en mallas, dos mallas con estructuras topológicas equivalentes, son superpuestas sobre una imagen. La especificación de la malla es directamente dependiente de la estructura de descomposición del dominio: descomposiciones bien estructuradas son fáciles de ser manipuladas, en cuanto descomposiciones arbitrarias son más difíciles de representar y mantener.

- **Ventajas en la especificación con mallas:** Las mallas regulares pueden llegar a simplificar grandemente la manipulación de datos en la implementación, también facilita la comprensión del proceso de transformación (es decir es más intuitivo). Es fácil de ajustar los elementos correspondientes de las mallas para indicar la deformación deseada.
- **Desventajas en la especificación con mallas:** A pesar que las mallas permiten una implementación extremadamente eficiente, también puede ser restrictivo ya que posicionar los puntos de la malla sobre las características interesantes de la imagen es una tarea difícil, y la regularidad de la malla no necesariamente coincide con la estructura "natural" de la imagen que ella cubre. Otro problema muy frecuente se da cuando la transformación modifica, áreas cercanas a la vecindad que se desea transformar. Así se ve en la figura.

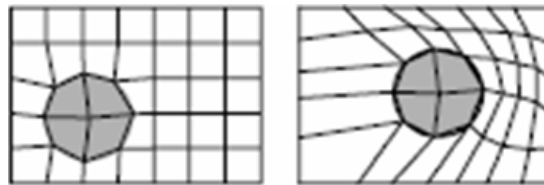


Figura 6: problema de la especificacion con malla.

4.1.2. Especificación por Características.- Es sabido que en una Especificación por partición define funciones de mapeo para todo el dominio, en cuanto en una especificación por características hace una función de mapeo indefinida en $U \rightarrow U_s$.

En este tipo de especificación de warping, algunas características especiales de la transformación son especificadas por el usuario. La deformación será computada de forma que se debe mapear cada característica de la imagen para su estado transformado correspondiente. Un caso importante de warping basado en características es el de la especificación basada en puntos donde cada característica es indicada por un punto en la imagen (Gomes, 1994).

- **Ventajas en la especificación por características:** la ventaja obvia de la especificación por características es que el usuario precisa especificar solo algunas características relevantes.

- **Desventajas en la especificación por características:** sin embargo es necesario frecuentemente adicionar algunas características secundarias para obtener resultados deseados, ya que las desiciones de interpolación tomadas por los algoritmos de deformación son hasta cierto punto arbitrarias.

En general, algunas condiciones útiles son exigidas de la especificación para que resulte en deformaciones razonables.

4.1.3. Implementación - Algoritmos Warping: Como se mencionó en el inicio de esta sección ahora se definirá la parte del cómo obtener los resultados que se plantearon con la especificación, para ello se presentará algunos algoritmos encontrados en la literatura sobre warping.

- **Warping por Malla de Triángulos:** Una manera simple de especificar un warping genérico de imágenes es usar mallas de triángulos. Un algoritmo para implementar deformaciones de este tipo es usar coordenadas baricéntricas, de tal forma que cualquier triángulo pueda ser mapeado en cualquier otro triángulo usando una transformada lineal.

Una rasterización de cada triángulo en la malla de destino acoplada con el mapeamiento mencionado, produce un mapeamiento inverso para deformar una imagen.

Un punto importante a resaltar es que el cambio de coordenadas es afectado por un mapeamiento lineal por partes, que es lineal en el interior de cada triángulo, pero tiene algunas continuidades C^0 en las aristas. De esta forma surgen algunos efectos indeseables durante el warping que será inevitable, como los mostrados en la figura (Gomes, 1994).

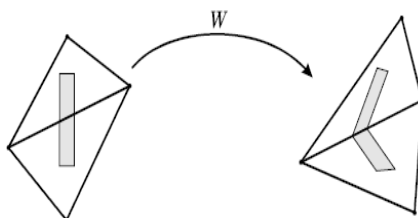


Figura 7: Discontinuidad del mapeamiento por coordenadas baricéntricas.

Entre los muchos problemas que se abordan está en conseguir una especificación conveniente y fácil de usar, especialmente en relación a mantener la consistencia de la representación de la malla durante su creación. La construcción de toda la malla, triángulo por triángulo, es un proceso largo y tedioso que puede ser tornado más fácil a través del uso de alguna forma de triangulación automática.

- **Warping por Malla de Splines en dos pasos:** Este algoritmo recibe como entrada dos mallas de splines indicando los estados normal y deformado de la imagen. El uso de splines para especificar las mallas regulares que definen el sistema de coordenadas presenta diversas ventajas. En primer lugar, el uso de una estructura regular, en vez de una topología arbitraria, simplifica significativamente tanto las

estructuras de datos como las funciones de mapeamiento. Puede ser demostrado que este mapeamiento es separable, i.e., puede ser realizado en dos pasos independientes, un horizontal, que produce una imagen intermedia horizontalmente deformada, y una vertical, que usa esta imagen intermedia para producir la salida final. Esto transforma el problema de deformar una imagen 2D en un problema unidimensional, reduciendo enormemente la complejidad de los cálculos (Gomes, 1994).

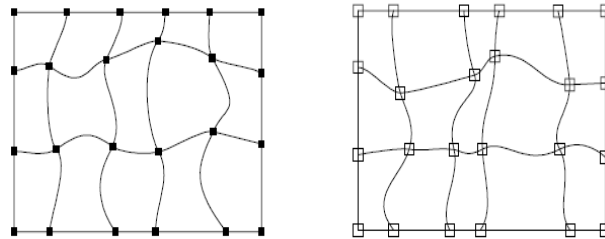


Figura 8: Mallas de splines: Origen - Destino.

El paso horizontal del algoritmo se inicia por la creación de una malla intermedia I que incluye solamente los desplazamientos horizontales de la malla origen S para la malla destino D, esto es, cada punto en I tiene la misma coordenada x que el punto correspondiente en D, y la coordenada y del punto correspondiente en S.

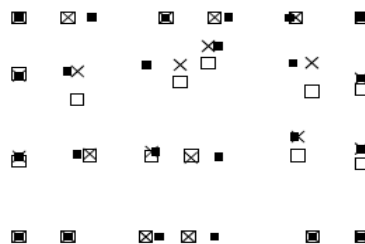


Figura 9: Puntos de control de las mallas origen, destino e intermedia.

Una spline interpolante es pasada por cada columna de S e I, produciendo S_s y I_s , S_s es el conjunto de splines verticales en S y I_s , el conjunto de splines verticales en I. Cada línea horizontal es entonces interceptada independientemente con S_s y I_s . las coordenadas x de las intersecciones con I_s y las coordenadas x de las intersecciones con S_s son entonces interpoladas con una nueva spline, la que resulta en la función de mapeamiento para cada línea. Para cada píxel de cada línea horizontal de la imagen intermedia, es ahora fácil de usar esta función de mapeamiento para determinar que píxeles de la imagen de entrada influyen este píxel de la salida.

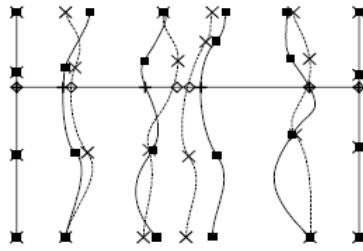


Figura 10: Splines verticales interceptadas con una scanline particular.

El paso vertical usa a malla intermedia y la imagen intermedia como entrada para producir a salida final; la malla origen y la imagen original no son usadas. Este paso es completamente análogo al primer paso, con las líneas de I y D interpoladas por esplines horizontales, y las funciones de mapeamento computadas para cada línea vertical

5. Aplicación de Transformada Warping para Simulación de Cirugía Plástica

En este trabajo se presenta dos *implementaciones*, i.e. algoritmos warping (funciones de mapeamento), los cuales se clasifican, según el fundamento de la sección anterior, dentro de la *especificación por particiones*. El objetivo de presentar dos técnicas diferentes es, comparar las propuestas, también con los algoritmos descritos en la sección anterior; y así poder determinar la mejor solución para simular los resultados obtenidos en una cirugía plástica estética. Las propuestas se describen en las dos secciones siguientes:

6. Warping basado en una Transformación a Coordenadas Polares

Como su nombre lo indica esta técnica esta basada en una transformación de coordenadas cartesianas a coordenadas polares (Bourke, 2002), esta técnica es muy útil para simular resultados de un aumento de determinadas zonas, esto lo podemos ver mejor en la Figura 11, y para nuestra aplicación se guía de los siguientes pasos:

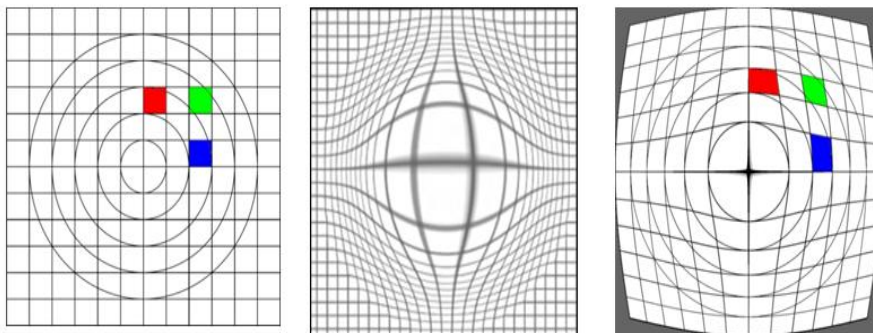


Figura 11: Warping basado en transformación a coordenadas polares.

- Para cada pixel (i,j) de la imagen (normalizar entre -1 y 1):

$$x = \frac{2i}{width} - 1 \quad y = \frac{2j}{height} - 1 \quad (4)$$

Donde: width, height son dimensiones de la imagen.

- Transformando las coordenadas cartesianas a polares (r,φ):

$$r = \sqrt{x^2 + y^2} \quad \phi = \text{atan2}(y, x) \quad (5)$$

- Aplicando la transformación T:

$$(x', y') = T(x, y) \quad \text{Donde: } T = (x, y) / x = r^e \cos(\phi), y = r^e \sin(\phi) \quad (6)$$

Y $e = 0,9$ (e es un valor obtenido experimentalmente).

- Transformando coordenadas polares a cartesianas:

$$i' = \frac{2 * width}{x'} - 1 \quad j' = \frac{2 * height}{y'} - 1 \quad (7)$$

Donde finalmente (i', j') son los valores que se reemplazaran por los píxeles de entrada(i,j).

7. Warping basado en superficies producto tensorial de Bezier

Se escoge el uso de mallas para la implementación de una transformación warping, por las ventajas descritas anteriormente, la fácil manipulación de gran cantidad de datos en la implementación, también facilita la comprensión del proceso de transformación (es decir es más intuitivo). Otra ventaja se da por el lado de superficies tensorial de Bézier, facilitando la construcción de malla y permitiendo realizar deformaciones eficientes por la simple variación del los puntos de control.

7.1. Definiciones previas

Se describe conceptos importantes que nos ayudarán a comprender, la construcción de las mallas que se usarán.

7.1.1. Polinomios de Bernstein: Los polinomios de Bernstein de grado n , que denotamos por: $B_{0,n}(t), B_{1,n}(t), \dots, B_{n,n}(t)$

$$B_{i,n}(t) = \binom{n}{i} (1-t)^{n-i} t^i \quad \text{para } i = 0, 1, \dots, n \quad (8)$$

Donde:

$$\binom{n}{i} = \frac{n!}{i!(n-i)!} \quad (9)$$

7.1.2. Curvas de Bezier: Se denomina curva Bezier asociada a $n + 1$ puntos P_0, P_1, \dots, P_n de \mathbb{R}^2 , a la curva parametrizada, definida para $t \in [0, 1]$, cuyos puntos vienen dados mediante la siguiente expresión (Paluszny, 2002).

$$(x(t), y(t)) = \sum_{i=0}^n B_{i,n}(t)P_i \quad (10)$$

en donde $B_{i,n}$ son los Polinomios de Bernstein de grado n .

Los puntos P_0, P_1, \dots, P_n que determinan una curva de Bézier se denominan puntos de control, y la poligonal que los une es el polígono Bézier o B-polígono.

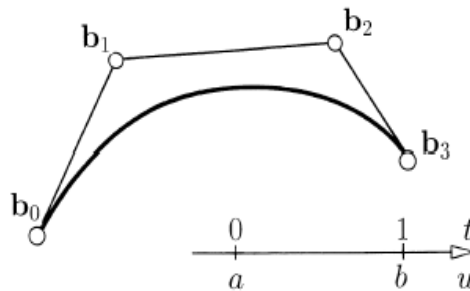


Figura 12: Curva cúbica con un polígono de bezier sobre $[a, b]$.

Las propiedades de las curvas de Bezier heredan las propiedades de los polinomios de Bernstein, describiremos las más útiles.

- **Propiedad de la envolvente convexa:** Cada curva Bézier se encuentra en la envolvente convexa de sus puntos de control, parte del punto P_0 y llega al punto P_n .
- **Simetría:** Cuando en una curva Bézier se cambia el parámetro t por $1 - t$, la curva se recorre en sentido contrario.

7.1.3. Superficies Producto Tensorial de Bezier: La manera más sencilla de construir una superficie es barrer una curva en el espacio, por ejemplo en la representación de Bézier. Los puntos de control de esta curva se mueven a su vez siguiendo curvas de Bezier, estos puntos definen la superficie. Por esta razón se puede trabajar con estas superficies aplicando los algoritmos para curvas. Análogamente se obtiene mallas de control con propiedades similares de las representaciones de curvas (Paluszny, 2002).

Para mostrar cómo se construye una superficie producto tensorial a partir de curvas de Bézier, sean

$$A_0(u), \dots, A_m(u) \text{ y } B_0(v), \dots, B_n(v) \quad (11)$$

dos conjuntos de Polinomios de Bernstein de grado m y n respectivamente y consideramos la curva

$$p(u) = \sum_{i=0}^m a_i A_i(u) \quad (12)$$

Cada uno de los puntos de control de esta curva yace sobre una curva de Bézier

$$a_i = a_i(u) = \sum_{j=0}^n b_{i,j} B_j(v) \quad (13)$$

Entonces la superficie p dada por

$$b(u, v) = \sum_{i=0}^m \sum_{j=0}^n b_{i,j} A_i(u) B_j(v) \quad (14)$$

Se denomina superficie producto tensorial de Bezier.

En notación mas compacta

$$b(s) = \sum_i b_i B_i^m(s) \quad (15)$$

donde $i = (i, j)$

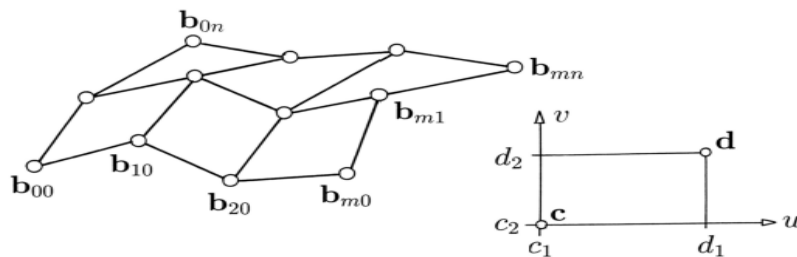


Figura 13: Una malla de Bezier

Propiedades importantes de las mallas de Bézier

Como en el caso de las curvas, las superficies tensoriales de Bézier heredan las propiedades de los polinomios de Bernstein (Paluszny, 2002)

- La simetría tiene una gran utilidad práctica consiste en que para parametrizar el mismo parche tomando como origen otro vértice, simplemente se invierte el orden de todas las filas o columnas o de ambas.
Como una curva pasa por su primer y último punto de Bézier se tiene que:
El borde de la malla de Bézier determina las cuatro curvas fronteras del parche $b[c,d]$.
En particular resulta, que las cuatro esquinas de parche y de la malla de Bézier coinciden, esto es: $b(c_1, d_1) = b_{00}$, $b(c_1, d_2) = b_{0n}$, etc.
- El parche $b[c, d]$ esta contenido en la cápsula convexa de sus puntos de Bézier.

7.2. Construcción de la malla producto tensorial de Bezier para Warping

Aprovechando las propiedades de las mallas de Bezier se construirá deformaciones específicas para cada necesidad, es decir para simular una rinoplastia se usará una deformación diferente, que para el caso de simular el aumento de medidas o liposucciones. Construcción de deformaciones para cada necesidad:

- **Aumento de medidas:** Esta deformación se usa para: el aumento de busto, glúteos, masa muscular, etc.

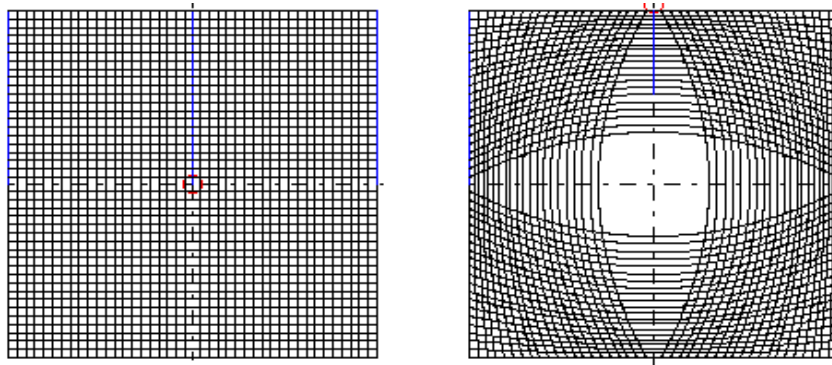


Figura 14: Malla sin deformar - Malla deformada(aumento).

- **Reducciones y Aumentos suaves:** Estas deformaciones se usan para: liposucción, rinoplastia, etc.

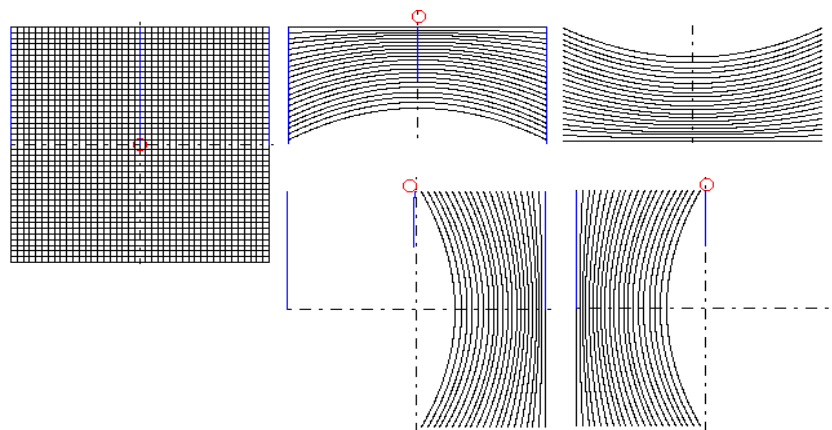


Figura 15: Malla sin deformar - Malla deformada(reducción y aumento Horizontal y Vertical).

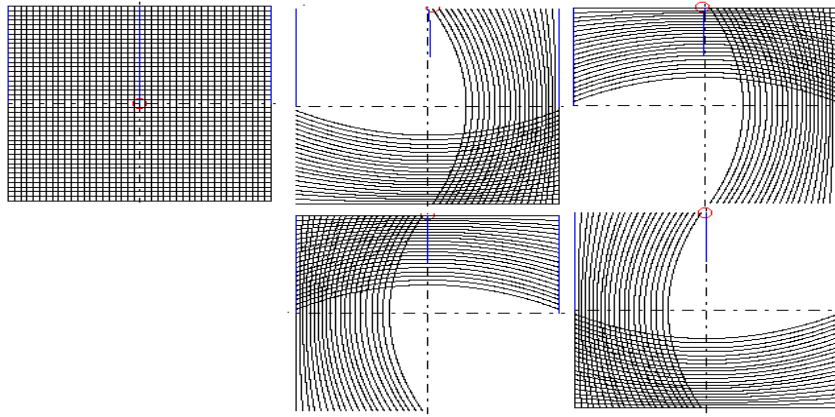


Figura 16: Malla sin deformar - Malla deformada(reducciones y aumentos oblicuos).

7.3. Proceso de Deformación de la Imagen

- 1.- Seleccionar el área que se desea deformar, esto según la transformación que se desea hacer(Rinoplastía,Liposucción,etc.)
- 2.- Sea I de orden $m \times n$ la subimagen que se desea deformar.
- 3.- Las curvas base para la construcción de la malla se formarán según:
 Curva Base 1 con puntos de control: $(0,0),(m/2,0),(m-1,0)$.
 Curva Base 2 con puntos de control: $(0,n/2),(m/2,n/2),(m-1,n/2)$.
 Curva Base 3 con puntos de control: $(0,n-1),(m/2,n-1),(m-1,n-1)$.
 Estas tres curvas se construirán simultáneamente de forma vertical en la imagen I.
- 4.- Por cada punto que describen las 3 curvas base se generan 3 puntos, los cuales serán puntos de control para cada curva horizontal. Aprovechando la propiedad de simetría de las curvas de Bezier y por ser un modelo simétrico, se construirá 3 curvas adicionales que serán simétricas a las primeras curvas ya creadas. Este paso se repite para ir creando la malla de acuerdo a la ecuación:

$$b(s, t) = \sum_{i=0}^m \sum_{j=0}^n b_{i,j} B_i^m(s) B_j^n(t) \quad (16)$$

Donde s recorrerá de 0 a 0.5 para cada curva vertical y t recorre de 0 a 1 para cada curva horizontal.

- 5.- Mover los píxeles según la posición que describa cada curva de la malla deformada.

Observación: La malla formada describe un conjunto de curvas que representan tanto las filas, como las columnas de la subimagen I, además el orden de la submatriz I debe ser cuadrada. El punto de control central que deformará toda la malla es el punto $(m/2,n/2)$, el cual se cambiará según la deformación que se desee. El orden del polinomio que se describen las curvas es cuadrático. Aquí se muestran algunos ejemplos sobre la aplicación del proceso de deformación de las mallas en las imágenes.

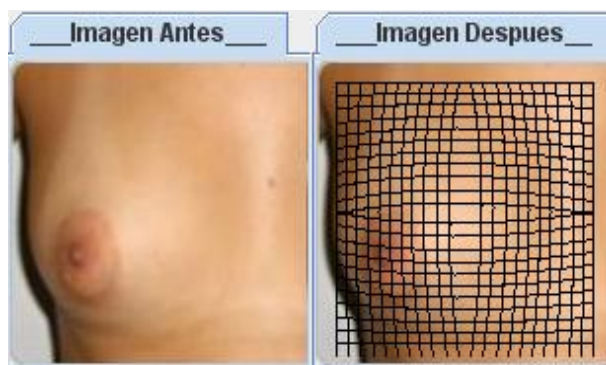


Figura 17: Imagen Inicial - Malla para Aumento de seno

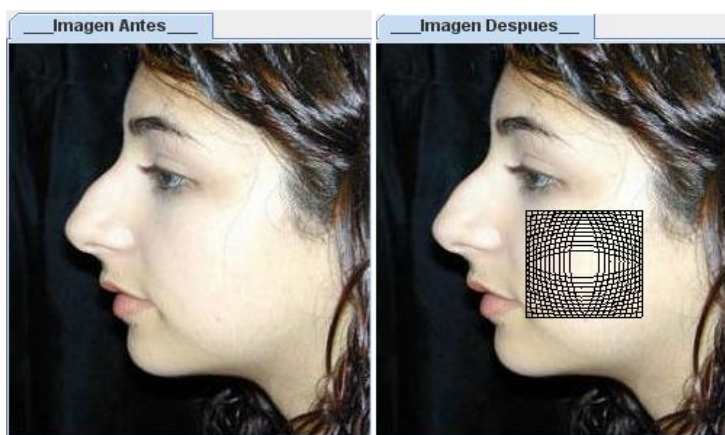


Figura 18: Imagen Inicial - Malla para aumento en el rostro

8. Algoritmo General

La forma de aplicar la técnica del Warping sobre una imagen, tanto con transformación a coordenadas polares como con superficies de Bezier, esta plasmado en el algoritmo 1, donde la entrada es una imagen I que se encuentra en formato RGB, y la salida deseada es una imagen I' , tambien en RGB, la cual muestra la simulación de cirugía plástica. Este algoritmo obtiene una submatriz I_0 de la imagen de entrada I cuyo tamaño depende de un radio, luego aplica la transformación warping sobre la submatriz I_0 , y al resultado de esta transformación se le aplica un filtro F (filtro definido experimentalmente para la aplicación) para suavizar I_0 . Finalmente se reemplaza I_0 en I , en la misma posición que se encontraba inicialmente.

Algoritmo 1 Transformación Warping para la Simulación de Cirugía Plástica

Requiere: imagen I en (R,G,B)

Asegurar: imagen I' en (R,G,B) (la cual simula la cirugía plástica aplicada a la imagen de entrada I)

Obtener una submatriz I_0 de I

Aplicar la Transformación Warping para cada pixel de I_0

Aplicar un filtro F para mejorar I_0

Reemplazar I_0 en I

9. Resultados

Este trabajo fue implementado sobre el lenguaje multiplataforma JAVA, bajo la IDE Net-Beans 6.1, En la aplicación abrimos una imagen para simular la cirugía plástica, luego es aplicado la transformación en la imagen mediante clic's, el tamaño de la submatriz dependerá del radio que sea asignado en la aplicación. Los resultados obtenidos son los siguientes:

- Rinoplastia:

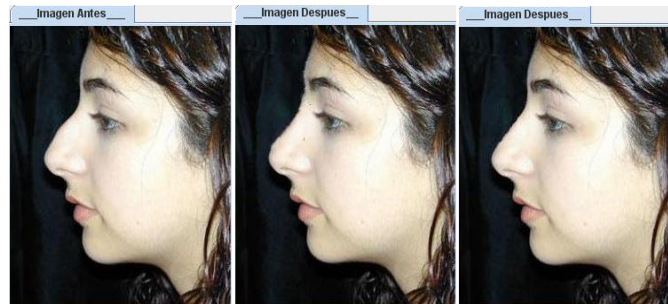


Figura 19: Simulación de Rinoplastia - Imagen Original, Resultado con Transformación a Coordenadas Polares, Resultado mediante Superficies Producto Tensorial de Bezier

- Aumento de Medidas:



Figura 20: Simulación de Aumento de Medidas (Senos) - Imagen Original, Resultado con Transformación a Coordenadas Polares, Resultado mediante Superficies Producto Tensorial de Bezier

- Reducción de Papada:



Figura 21: Simulación de Reducción de Papada - Imagen Original, Resultado con Transformación a Coordenadas Polares, Resultado mediante Superficies Producto Tensorial de Bezier

- Aumento de masa Muscular:



Figura 22: Simulación de Aumento de Masa Muscular - Imagen Original, Resultado con Transformación a Coordenadas Polares, Resultado mediante Superficies Producto Tensorial de Bezier

- Liposucción:

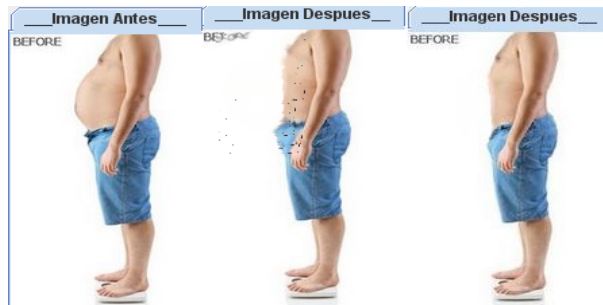


Figura 23: Simulación de Liposucción - Imagen Original, Resultado con Transformación a Coordenadas Polares, Resultado mediante Superficies Producto Tensorial de Bezier

- Rinoplastía y Reducción de Papada:

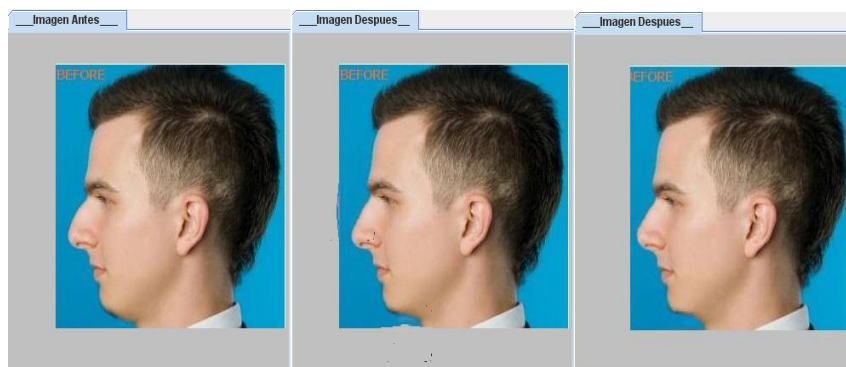


Figura 24: Simulación de Rinoplastía y Reducción de Papada - Imagen Original, Resultado con Transformación a Coordenadas Polares, Resultado mediante Superficies Producto Tensorial de Bezier

- Comparación con Deformación obtenida por el Software GIMP:

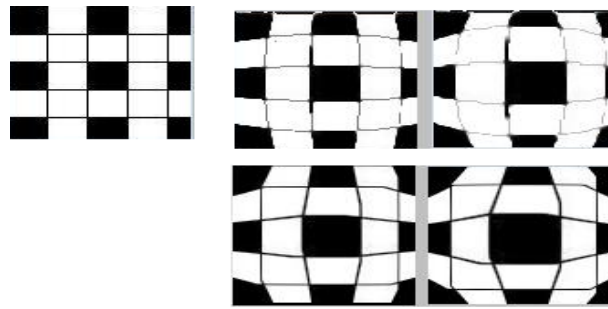


Figura 25: (superior-izquiero)Imagen Original, (superior-derecho)Resultado mediante Superficies Producto Tensorial de Bezier,(inferior-derecho) Resultado obtenido con el software GIMP

- Comparación con Deformación obtenida por el Software Face Touch Up:

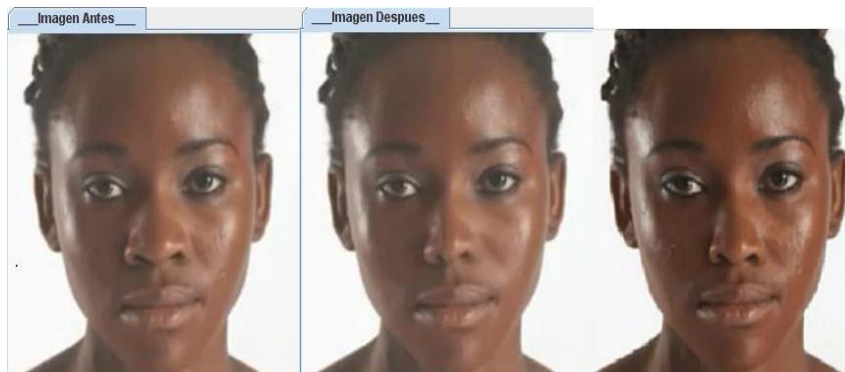


Figura 26: Imagen Original - Resultado con nuestra aplicación mediante mallas - Resultado con el software Face Touch Up.

10. Discusión de los Experimentos

Los resultados obtenidos son muy satisfactorios en cuanto a la calidad de la imagen, esto es porque en la primera técnica (usando coordenadas polares) se pierde información de los bordes de la subimagen lo cual si no es interpolado correctamente genera pequeñas manchas negras en los resultados. Se debe tener en cuenta que nuestra aplicación es interactiva con el usuario, por lo cual un resultado satisfactorio es dependiente de este (cirujano - paciente), el cual puede realizar múltiples deformaciones con radios variables. En la figura 25 se muestra una comparación con el GIMP (software libre para la edición de imágenes), en el cual podemos ver que el resultado obtenido mediante superficies de Bezier mantiene la forma de cada rectángulo de la imagen, mientras que en el GIMP estos se deforman. Para una simulación completa de cirugía plástica lo que también se debe incluir es la corrección de arrugas y manchas, para esto también se puede hacer uso de la técnica ya mencionada, pero con una función adecuada a las necesidades de la aplicación.

11. Conclusiones

Existen algunos trabajos como los mencionados en la sección 2 que explotan esta técnica en una aplicación real, en este trabajo se muestra como la técnica del Warping, puede ser usada de manera local definiendo una vecindad, para una aplicación real.

También debemos recalcar que durante la investigación, se ha resuelto el problema de la especificación mediante mallas, planteada en el punto 3.1.1.

La complejidad de la aplicación depende de la función de mapeamiento que se defina, en el caso de la transformación mediante las superficies de Bezier es eficiente, ya que se utiliza curvas de Bezier de segundo grado para la construcción de la malla, y en la transformación mediante las coordenadas polares esta eficiencia dependerá de la función vista en la sección 6.

Luego de ver la aplicación de estas 2 técnicas para realizar el Warping sobre una imagen, comprobamos que con la aplicación de superficies producto tensorial de Bezier se obtienen mejores resultados, en cuanto a la calidad de la imagen, que con la aplicación de transformación a coordenadas polares. Algunas técnicas parecidas al Warping son el zoom localizado, pero en la práctica se demuestra que el Warping produce una deformación, más suavizada.

Actualmente existen múltiples aplicaciones en las cuales se puede aplicar la técnica Warping, por ejemplo en el procesamiento del sonido es usada para la mezcla y la deformación de sonidos en general, para video es usada para recrear animaciones en base a imágenes estáticas, y en imágenes como en este trabajo es usada para realizar deformaciones.

Referencias

- Bourke, P. (2002). Image warping / distortion. <http://local.wasp.uwa.edu.au/~pbourke/miscellaneous/imagewarp/>.
- Gomes, J. (1994). Deformação e metamorfose de objetos gráficos. *Pontifícia Universidade Católica do Rio de Janeiro*.
- Heckert, P. (1989). Fundamentals of texture mapping and image warping. *University of California, Berkeley*.
- Kaeria, S. (2004). Vpss - virtual plastic surgery software. <http://www.photo-warping.com/>.
- Lizardo, R. (2007). Procesamiento digital de imágenes. UDEC.
- Paluszny, M. (2002). Métodos de Bezier y b-splines. Universidad Karlsruhe- Editorial Springer.
- Physicians, W. U. (2009). Face touch up. <http://www.facetouchup.com>.

Lista de Autores

Albornoz, L., 57
Alegría, D., 1
Arauco, S., 67
Arce, P., 1
Arteaga, J. F., 77
Atoche, L., 49

Cervantes, K., 67
Chevarría, G. M. , 49
Colán, J., 67

Escovedo, T., 31

Fonseca, P., 21

Grández, W., 49
Guevara, J. L., 39

Jave, L., 67
Jiménez, L., 1

Lujan, L. A., 11
Luna, M., 21

Melo, R. N., 31

Obando, S. F., 77

Rocca, J., 77
Rodríguez , E., 11

Sánchez, H. Y., 11
Salazar, J. O., 39

Trelles, J., 21

Vidal, C., 1, 57
Vidal-Duarte, E., 57

Zuñiga, A., 57

Organizadores



SOCIEDAD
PERUANA DE
COMPUTACIÓN



Auspiciadores

