



UNIVERSIDADE DA CORUÑA

FACULTADE DE INFORMÁTICA

TRABALLO FIN DE GRAO

GRAO EN ENXEÑARÍA INFORMÁTICA

Mención en Enxeñaría do Software

**Aplicación web para a análise do  
comportamento humano en zoas transitadas**

**Autor:** Xoan Iago Suárez Canosa

**Director:** Cancela Barizo, Brais

**Director:** González Penedo, Manuel Francisco

**Director:** Novo Buján, Jorge

**Director:** Ortega Hortas, Marcos

A Coruña, 8 de xuño do 2015



*A Meus pais Elvira e Manolo, que grazas ao seu  
esforzo e cariño permitíronme chegar ata aquí*



# Agradecementos

En primeiro lugar grazas a Brais Cancela por toda a axuda e o apoio que me brindou ao largo destes meses, xa que sen a súa achega este proxecto non tería nin sequera comezado. Grazas tamén a tódolos compañeiros da facultade que coñecín ao largo deste increíbles anos, deles aprendín a meirande parte do que sei e ademais pasamos momentos inesquecibles. Por último agradecer á miña familia e ao resto dos meus amigos por soportar día a día cada un dos meus defectos e permitirme gozar das vosas virtudes.



# Resumo

No mundo da seguridade, un dos maiores retos que se propoñen hoxe en día é a detección de condutas sospeitosas. Esta situación vólvese cada vez máis complexa ao aumentar o número de cámaras a vixiar, polo que é imprescindible dispoñer dunha ferramenta que facilite esta tarefa.

Este proxecto consiste nunha aplicación web que emprega funcionalidades relativas á análise do comportamento en zoas transitadas, incluíndo visualización de vídeo e distintas capas que mostran información de alto nivel sobre o comportamento detectado.

En concreto esta ferramenta encárgase de detectar a todos aqueles obxectos ou persoas que aparecen nunha secuencia de vídeo, aplicando un algoritmo para medir o 'grao de anormalidade' da súa conduta en base aos movementos que realizan.

## Palabras chave

Análise Comportamento, Comportamento Humano, Aplicación Web, Django.





# Índice xeral

|   |           |
|---|-----------|
| <b>1. Introducción e Obxetivos</b>  | <b>1</b>  |
| <b>2. Fundamentos Teóricos e Conceptos Previos</b>                                    | <b>3</b>  |
| 2.1. Programación Web . . . . .   | 3         |
| 2.2. OpenCV . . . . .   | 3         |
| 2.3. Análise do Comportamento . . . . .   | 3         |
| <b>3. Análise de antecedentes e alternativas</b>                                      | <b>5</b>  |
| <b>4. Metodoloxía seguida no desenvolvemento de Proxecto</b>                          | <b>7</b>  |
| 4.1. Persoas . . . . .  | 7         |
| 4.2. Reunións . . . . .   | 8         |
| 4.3. Control de Versións con GitHub . . . . .   | 9         |
| 4.3.1. Título . . . . .   | 9         |
| 4.4. Integración Continua con TravisCI . . . . .                                      | 9         |
| 4.5. Control da cobertura con Coveralls . . . . .                                     | 9         |
| 4.6. Xestión de Incidencias e Control de Proxecto con YouTrack . . . . .              | 9         |
| <b>5. Estudo das tecnoloxías para a creación de aplicacións web baseadas en vídeo</b> | <b>11</b> |
| 5.1. Back-End . . . . .   | 12        |
| 5.1.1. Java . . . . .   | 12        |
| 5.1.2. C# . . . . .   | 12        |
| 5.1.3. C - C++ . . . . .  | 12        |

|  |           |
|--|-----------|
| 5.1.4. Python + Django . . . . .               | 13        |
| 5.2. Front-End . . . . .                       | 13        |
| 5.2.1. Vídeo en Flash . . . . .                | 13        |
| 5.2.2. Vídeo HTML5 + Javascript . . . . .      | 14        |
| 5.3. Título sección . . . . .                  | 15        |
| 5.3.1. Título subsección . . . . .             | 15        |
| 5.3.2. Título subsección . . . . .             | 16        |
| <b>6. Funcionalidades Destacadas</b>           | <b>19</b> |
| 6.1. Control de Usuarios . . . . .             | 19        |
| 6.2. Reprodución de Vídeo . . . . .            | 19        |
| 6.3. Carga de Vídeo . . . . .                  | 19        |
| 6.4. Análise de Vídeo . . . . .                | 19        |
| 6.5. Detección de Obxectos . . . . .           | 19        |
| 6.6. Traxectorias . . . . .                    | 19        |
| 6.7. Comportamento anormal . . . . .           | 19        |
| 6.7.1. Título . . . . .                        | 19        |
| <b>7. Probas Realizadas</b>                    | <b>21</b> |
| 7.1. Probas Unitarias . . . . .                | 21        |
| 7.1.1. Probas Automáticas . . . . .            | 21        |
| 7.2. Probas de Integración . . . . .           | 21        |
| 7.2.1. Probas funcionais Selenium . . . . .    | 21        |
| 7.3. Probas de Sistema . . . . .               | 21        |
| 7.4. Probas de Aceptación . . . . .            | 21        |
| <b>8. Planificación e Avaliación de Custes</b> | <b>23</b> |
| <b>9. Resultados e Conclusións</b>             | <b>25</b> |
| <b>10. Liñas Futuras</b>                       | <b>27</b> |
| 10.1. Vídeo en Directo . . . . .               | 27        |

---

|   |           |
|---|-----------|
| <b>A. Título</b>                              | <b>29</b> |
| A.1. Lista de Acrónimos . . . . .             | 29        |
| A.2. Manual de Usuario . . . . .              | 29        |
| A.3. Manual de referencias Técnicas . . . . . | 29        |
| A.4. Notas acerca da Terminoloxía . . . . .   | 29        |



# Índice de figuras

|   |    |
|---|----|
| 5.1. Esta es la etiqueta de la figura . . . . .           | 16 |
| 5.2. Esta es la etiqueta de la figura con borde . . . . . | 16 |



# Índice de Táboas

|                                 |    |
|---------------------------------|----|
| 5.1. Tabla de ejemplo . . . . . | 17 |
|---------------------------------|----|





# Capítulo 1

## Introducción e Obxetivos

Existen distintos métodos para o ana...



## Capítulo 2

# Fundamentos Teóricos e Conceptos Previos

En este capítulo se exponen los fundamentos teóricos en los que se basa el proyecto o que se utilizan en él. Normalmente cada proyecto trata de una temática, de la que hay una serie de conocimientos teóricos relacionados y que deben incluirse en la memoria del proyecto, para aportarle el carácter científico que conlleva todo trabajo académico. Se requiere una explicación detallada de todo lo necesario, para dar una visión profunda de lo que debe conocerse para afrontar la realización del proyecto. Por una parte sirve como exposición de los elementos científicos en los que se basa el proyecto, pero a la vez, como explicación de lo que se ha estudiado para la elaboración del mismo.

### 2.1. Programación Web

### 2.2. OpenCV

### 2.3. Análise do Comportamento



## Capítulo 3

# Análise de antecedentes e alternativas

Se trata de realizar un estudio de alternativas o “estado del arte” o un análisis comparativo de alternativas.

Se exponen las diferentes alternativas que se han evaluado o que se consideran de interés, a lo realizado en el proyecto. Fundamentalmente se trata de otras herramientas existentes que realizan algo similar, sean o no comerciales, o de prototipos de investigación relacionados, o de estudios que tratan aspectos similares.

Buscar por internet produtos que fagan algo similar...



## Capítulo 4

# Metodoloxía seguida no desenvolvemento de Proxecto

Os diferentes obxectivos do proxecto abordáronse seguindo a Metodoloxía SCRUM, adaptada a un proxecto de un solo Developer.

Esta metodoloxía áxil tamén chamada melé pola súa inspiración no Rugby, permite un desenvolvemento rápido en situacións de requisitos inestables. Apoiase no seu carácter iterativo e incremental, dividindo traballo a realizar en períodos de aproximadamente un mes chamados Sprint's.

Para a realización deste traballo de fin de grao foi preciso adaptala, pois está pensada en principio para organizar equipos de entre 3 a 9 persoas (Team). Por outro lado, o marco de traballo planifica reunións diarias (Daily Scrum), ao supoñer que todos os membros do equipo traballan unha xornada laboral enteira entre cada unha destas reunións, o cal tampouco se dá no caso deste proxecto, xa que a dedicación será de determinadas horas nos momentos dispoñibles.

### 4.1. Persoas

Os tres papeis que se definen nesta metodoloxía [1] foron adaptados do seguinte xeito:

- **ProductOwner:** O papel de ProductOwner, que define os requisitos da aplicación estivo representado polo director de proxecto Brais Cancela, que participou na creación do Anteprojecto. En certos momentos o señor Cancela tamén desempeñou a función de membro do equipo, posto que é foi autor do algoritmo de análise de vídeo.
- **ScrumMaster e Development Team:** Ambos papeis leváronse a cabo polo autor, xa que carece de sentido definir ambas figuras nun equipo de unha soa persoa. De este xeito á par que se desenvolvía o proxecto, íase asegurando o cumprimento das regras de SCRUM.

## 4.2. Reunións

As reunións pola súa parte modifícanse do seguinte xeito:

- **Sprint Planning Meeting:** Esta reunión mantén o mesmo formato que no SCRUM puro, xuntando ao autor co ProductOwner e concretando as tarefas do Product Backlog que se realizarán no seguinte Sprint, pasando por tanto a formar parte do Sprint Backlog.
- **Daily Scrum:** Dado que o equipo de Desenvolvemento e o ScrumMaster están conformados pola mesma persoa e que o número de horas diarias adicadas é moito menos ao dunha xornada laboral, considerouse oportuno substituír esta reunión diaria por unha reunión dúas veces á semana (Martes e Xoves pola tarde normalmente). Na que se mostrase ao ProductOwner o avance do proxecto.
- **Sprint Review:** Esta reunión fusionase co Sprint Planning Meeting, xa que ao mesmo tempo valorase o traballo realizado no Sprint que remata e, en base a el, planifícase a videira Iteración.
- **Sprint Retrospective:** Pola súa parte, esta reunión toma un carácter unipersonal, pasando a ser unha valoración do propio autor sobre as persoas, relacións,



procesos e ferramentas implicadas no último Sprint. Nela avalíase os elementos con éxito e os suxeitos a melloras, creando un plan para implementar estas melloras na Videira iteración.

### **4.3. Control de Versións con GitHub**

Bla bla bla

#### **4.3.1. Título**

Bla bla bla

#### **Título**

Bla bla bla

### **4.4. Integración Continua con TravisCI**

### **4.5. Control da cobertura con Coveralls**

### **4.6. Xestión de Incidencias e Control de Proxecto con YouTrack**



## Capítulo 5

# Estudo das tecnoloxías para a creación de aplicacións web baseadas en vídeo

A continuación mostrase un estudo feito sobre as tecnoloxías mais empregadas hoxe en día de cara ao deseño de aplicacións web baseadas en vídeo. Os criterios que se valoraron á hora de escoller unha destas metodoloxías son os seguintes:

- **Plataforma e Portabilidade**

Segundo as especificacións iniciais do proxecto, e de cara a facilitar a implementación deste, as ferramentas empregadas deben de executarse baixo os Sistemas Operativos baseados en Linux.

- **Compatibilidade co algoritmo de Procesado de Vídeo**

Dado que o algoritmo que procesa o vídeo foi parcialmente implementado en c++, a tecnoloxía que se empregue para o desenvolvemento da parte web debe dispoñer da máxima compatibilidade con esta linguaxe de programación.

- **Desenvolvemento Áxil**

A tecnoloxía que se seleccione ten que minimizar o custe en tempo e esforzo da implementación, por este motivo valorarase positivamente que dispoña de IDE's

axeitados, facilidades de acceso a BD(Base de Datos),se é posible ORM(Object-Relational Mapping) integrado, recarga en quente... En xeral todo aquilo que permita axilidade e flexibilidade.

## 5.1. Back-End

O Back-End é a parte que procesa os datos proporcionados polo Front-End, atende as súas peticións, e xestiona o modelo de datos e os procesos implicados na aplicación. Neste caso en concreto, o Back-End será o encargado de interactuar directamente co sistema de Análise de Vídeo.

As tecnoloxías estudadas para esta parte do sistema son as seguintes:

### 5.1.1. Java

Java é unha das linguaxes mais empregadas actualmente. Ademais existen diversos frameworks web como Tapestry ou SpringMVC e en canto as BD Hibernate, que facilitan o seu uso, mais é preciso integralos xa que non forman parte da plataforma en si. A súas posibilidades de integración con c++ son altas grazas á interface JNI(Java Native Interface), pero a súa configuración pódese volver tediosa. Un dos proxectos estudados para ver o seu funcionamento é Red5 [2].

### 5.1.2. C#

esta linguaxe en combinación con .NET resulta unha combinación bastante áxil de cara á programación web, integrando na propia plataforma un deseñador Web e un ORM moi intuitivo. O gran problema polo que se descartou este entorno foi polo seu baixo grao de compatibilidade cos sistemas operativos Linux.

### 5.1.3. C - C++

C++ presenta como era de esperar a maior compatibilidade co algoritmo implementado, non obstante, inda que existen algunhas utilidades que facilitan o desenvolvemento

web con esta linguaxe como Wt (Web Toolkit)[3], o grado de axilidade está moi por baixo do que facilitan o resto das combinacións. Tamén pode resultar de interese o coñecido proxecto Icecast[4], que fai streaming de vídeo sobre unha interface web.

#### 5.1.4. Python + Django

Python presentase como a mellor opción para desenvolver o lado servidor, por unha parte dispón do módulo Subprocess[5] que permite executar calquera comando pola terminal maximizando así a modularidade e a integración co algoritmo en c++. Por outra parte Django[6] contén un potente ORM e un sistema de "Templates" que simplifica a parte web. Para concluír cabe destacar o feito de que python sexa unha linguaxe interpretada, xa que isto evita o paso previo de compilación.

## 5.2. Front-End

É a parte do Sistema encargada de interactuar co usuario. O maior requisito de esta parte é ser capaz de representar todas as capas de información complexa que xera o Back-End. E tratándose dunha aplicación web estudáranse aqueles xeitos que permitan a reprodución de vídeo, e a maiores o debuxado de figuras en movemento sobre este vídeo.

### 5.2.1. Vídeo en Flash

Video Flash é a tecnoloxía de reprodución de vídeo mais empregada e madura en internet dende hai anos. Inicialmente creada por Macromedia e mercada por Adobe en 2005, permite crear elaboradas animacións vectoriais, que logo poden proxectarse sobre un vídeo, e á vez tamén manexa os eventos de reprodución de vídeo como o play ou o stop.

Ten certos problemas en tanto ao Posicionamento Web(SEO), reprodución en dispositivos móbiles, accesibilidade... pero o meirande de todos eles é que mentres que outros dos exemplos estudados son 100 % libres, Flash é un programa propietario para

o que é preciso adquirir unha licenza.

### 5.2.2. Vídeo HTML5 + Javascript

Está é unha das combinacións mais empregadas actualmente, xa que segue o estándar do W3C (World Wide Web Consortium)[7]. Nel defínese como se han de mostrar e obter os vídeos dunha páxina codificada coa linguaxe HTML5, e destaca por seres extremadamente sinxelo en comparación con Flash ou outras tecnoloxías.

Para a modificación da súa aparencia pódese empregar *CSS3* (Follas de Estilos en Cascada), e para modificar o seu comportamento, o elemento `< video >` de HTML5 achega unha serie de Métodos, Eventos e Propiedades[8] que poden empregarse dende código *Javascript*.

Isto resulta de moita utilidade, xa que mediante Javascript tamén pode manexarse o elemento `< canvas >` de HTML5, que xera un mapa de bits para construír gráficos, manipular imaxes e crear dinamicamente animacións nunha páxina web. A única dúbida que soe xurdir sobre esta tecnoloxía está en canto ao seu rendemento e alcance, pero exemplos como os que amosa Kevin Roast na súa páxina web[9] despexan toda dúbida posíbel.

Tamén é de interese a librería de estilos web *Twiter Bootstrap* [10] para axilizar o deseño web con CSS3, Así como a librería de Javascript *JQuery* [11] que minimiza de xeito considerable as liñas de código a implementar.

Esta es una referencia a un artículo [12].

Esta es una referencia a un capítulo dentro de un libro [13].

Esta es una referencia a un libro [14].

Esta es una referencia a un artículo dentro de los proceedings de un congreso [15].

Esta es una referencia a una url [16].

**Esto está escrito en negrita**

*Esto está en enfatizado*

Este texto está centrado

Esto es una lista:

- Primer elemento
- Segundo elemento
- Lista dentro de otra lista:
  - Primer subelemento
  - Segundo subelemento

### 5.3. Título sección

Esto es una descripción:

**Palabra** descripción de la palabra

**Palabra** descripción de la palabra

Y esto una lista numerada:

1. Elemento
2. Elemento
3. Elemento
4. Elemento

Podemos incluir una figura y referenciarla de esta forma 5.1. Además podemos poner la página en la que está: 16

#### 5.3.1. Título subsección

Existen varias formas de incluir ecuaciones matemáticas. Las más utilizadas son las siguientes:



Figura 5.1: Esta es la etiqueta de la figura



Figura 5.2: Esta es la etiqueta de la figura con borde

- Para introducir expresiones matemáticas en el texto, se utiliza como delimitador el símbolo del dólar. Por ejemplo  $a \rightarrow b$ .
- Para introducir ecuaciones matemáticas, se utiliza el entorno `equation`:

$$\gamma = \frac{\bar{\alpha}}{\sqrt{\beta}} \quad (5.1)$$

$$E(v) = \int_0^1 \int_0^1 \int_0^1 E_{int}(v(r, s, t)) + E_{ext}(v(r, s, t)) dr ds dt \quad (5.2)$$

Un subíndice se especifica con el guión bajo y un superíndice con el circunflejo. Si el super/subíndice contiene varios caracteres, estos deben estar delimitados por llaves. Consultar el manual para comprobar como se pueden introducir símbolos y expresiones matemáticas en latex.

### 5.3.2. Título subsección

Esto es una subsección<sup>1</sup>.

Así introducimos texto sin ningún tipo de formato latex:

```
4 drwxr-xr-x  2 noelia imagen    4096 2005-09-12 12:09 figures
4 -rwxr--r--  1 noelia imagen    585 2005-09-12 16:56 Makefile
```

---

<sup>1</sup>Así se hace una nota a pie de página



```

4 -rw-r--r-- 1 noelia imagen 647 2005-09-12 17:38 memoria.aux
4 -rw-r--r-- 1 noelia imagen 1011 2005-09-12 17:18 memoria.bbl
4 -rw-r--r-- 1 noelia imagen 1171 2005-09-12 17:18 memoria.blg
16 -rw-r--r-- 1 noelia imagen 13440 2005-09-12 17:38 memoria.dvi
4 -rw-r--r-- 1 noelia imagen 412 2005-09-12 17:38 memoria.glg
4 -rw-r--r-- 1 noelia imagen 188 2005-09-12 17:38 memoria.glo
4 -rw-r--r-- 1 noelia imagen 241 2005-09-12 17:38 memoria.gls
4 -rw-r--r-- 1 noelia imagen 299 2005-09-12 17:38 memoria.ist
4 -rw-r--r-- 1 noelia imagen 283 2005-09-12 17:38 memoria.lof
12 -rw-r--r-- 1 noelia imagen 8997 2005-09-12 17:38 memoria.log
4 -rw-r--r-- 1 noelia imagen 87 2005-09-12 17:38 memoria.lot
108 -rw-r--r-- 1 noelia imagen 103492 2005-09-12 17:18 memoria.pdf
280 -rw-r--r-- 1 noelia imagen 278687 2005-09-12 17:18 memoria.ps
4 -rwxr--r-- 1 noelia imagen 1887 2005-09-12 17:40 memoria.tex
4 -rw-r--r-- 1 noelia imagen 735 2005-09-12 17:38 memoria.toc

```

### Título subsubsección

La figura 5.1 muestra un ejemplo de tabla básica. Para crear una tabla se utiliza el entorno `tabular` dentro del flotante `table`.

Tras `\begin{tabular}` se declara el número de columnas de la tabla. Cada columna se especifica con una letra:

**c** Si el texto en la columna está centrado

**l** Si el texto en la columna está alineado a la izquierda

**r** Si el texto en la columna está alineado a la derecha

Táboa 5.1: Tabla de ejemplo

|                        |                    |                                |
|------------------------|--------------------|--------------------------------|
| Fila 1 Col. 1 larálala | Fila 1 Col. 2 lara | Fila 1 Col. 3 lalalalalalalala |
| Fila 2 Col. 1          | Fila 2 Col. 2      | Fila 2 Col. 3                  |

Así `\begin{tabular}{rrrr}` indica 4 columnas con texto alineado a la derecha y `\begin{tabular}{cl}` indica dos columnas, la primera centrada y la segunda alineada a la izquierda.

Entre `\begin{tabular}{...}` y `\end{tabular}` se escribe el contenido de la tabla. Las columnas se separan con `&` y las filas con `\\`. Un ejemplo de una fila con tres columnas sería el siguiente:

$$\text{aaa} \ \& \ \text{bbbb} \ \& \ \text{cccc} \ \backslash\! \! \backslash^2.$$

Para incluir líneas horizontales existe el comando `\hline`. Tras `\begin{tabular}{...}` podemos incluir un `\hline`, pero **ojo**, en cada fila `\hline` siempre debe ir tras `\\`.

También es posible incluir líneas verticales, pero los manuales de estilo no lo aconsejan. Las líneas verticales se definen en la declaración con barras verticales. Por ejemplo, con `{|c|c|c}` se crearían 3 líneas verticales, una antes de la primera columna, otra entre la primera y la segunda columna, y la tercera, entre la segunda y tercera columna.

Dentro de las tablas podemos incluir expresiones matemáticas, texto enfatizado, negrita, etc. Es posible también fusionar varias filas o varias columnas dentro de una misma tabla. Para ello, consultar en manuales los comandos/paquetes `multirow` y `multicol`.

---

<sup>2</sup>Importante: debe coincidir el número de columnas en la declaración con el número de columnas que se escriben dentro de la tabla

## Capítulo 6

# Funcionalidades Destacadas

Bla bla bla

### 6.1. Control de Usuarios

### 6.2. Reproducción de Vídeo

### 6.3. Carga de Vídeo

### 6.4. Análise de Vídeo

### 6.5. Detección de Obxectos

### 6.6. Traxectorias

### 6.7. Comportamento anormal

Bla bla bla

#### 6.7.1. Título

Bla bla bla

**Título**

Bla bla bla

## Capítulo 7

# Pruebas Realizadas

### 7.1. Pruebas Unitarias

#### 7.1.1. Pruebas Automáticas

Estaría ben!

### 7.2. Pruebas de Integración

#### 7.2.1. Pruebas funcionais Selenium

### 7.3. Pruebas de Sistema

### 7.4. Pruebas de Aceptación



## Capítulo 8

# Planificación e Avaliación de Custes





## Capítulo 9

# Resultados e Conclusións

Podemos concluír que ...



## Capítulo 10

# Liñas Futuras

Aquí meter todas as tarxetas que queden pendentes + paridas varias...

### 10.1. Vídeo en Directo

Escribir aquí todo lo documentado del vídeo en streaming



# Apéndice A

## Título

Bla bla bla

### A.1. Lista de Acrónimos

Bla bla bla

- IDE
- BD

### A.2. Manual de Usuario

### A.3. Manual de referencias Técnicas

### A.4. Notas acerca da Terminoloxía

- Scrum Team
- Product Owner
- Development Team
- Scrum Master
- Spring

- Sprint Planning Meeting
- Sprint Goal
- Daily Scrum
- Sprint Review
- Sprint Retrospective
- Product Backlog
- Sprint Backlog

Se pueden escribir unas líneas en las que se justifique el motivo de la inclusión de términos en inglés, por ejemplo por formar parte de la jerga, ser de uso frecuente y estar completamente aceptados por comités científicos de la temática del proyecto, etc.; en otro caso deben buscarse traducciones adecuadas.

# Bibliografía

- [1] Schwaber, K. and Sutherland, J., *La Guía Definitiva de Scrum: Las Reglas del Juego*. Scrum.Org and ScrumInc, 2013.
- [2] “Red5 web server on github.” Available on: <https://github.com/Red5/red5-server>.
- [3] “Web toolkit for c++ page.” Available on: <http://www.webtoolkit.eu/wt>.
- [4] “Icecast server on github.” Available on: <https://github.com/paluh/icecast>.
- [5] “Python subprocess module.” Available on: <https://docs.python.org/3.4/library/subprocess.html>.
- [6] “Django web page.” Available on: <https://www.djangoproject.com/>.
- [7] “W3schools html5 video web page.” Available on: [http://www.w3schools.com/html/html5\\_video.asp](http://www.w3schools.com/html/html5_video.asp).
- [8] “Html audio and video dom reference.” Available on: [http://www.w3schools.com/tags/ref\\_av\\_dom.asp](http://www.w3schools.com/tags/ref_av_dom.asp).
- [9] “Kevin roast canvas examples.” Available on: <http://www.kevs3d.co.uk/dev/index.html>.
- [10] “Twitter bootstrap web page.” Available on: <http://getbootstrap.com/>.
- [11] “Página oficial de jquery.” Available on: <https://jquery.com/>.
- [12] Aamport, L. A., “The gnats and gnus document preparation system,” *G-Animal’s Journal*, 1986.

- [13] Knuth, D. E., *Fundamental Algorithms*, ch. 1.2. Addison-Wesley, 1973.
- [14] Knuth, D. E., *Seminumerical Algorithms*. Addison-Wesley, 1981.
- [15] Oaho, A. V., Ullman, J. D., and Yannakakis, M., “On notions of information transfer in VLSI circuits,” in *Proc. Fifteenth Annual ACM* (Oz, W. V. and Yannakakis, M., eds.), no. 17 in All ACM Conferences, (Boston), pp. 133–139, Academic Press, Mar. 1983. This is a full INPROCEEDINGS entry.
- [16] Comon, H., Dauchet, M., Gilleron, R., Jacquemard, F., Lugiez, D., Tison, S., and Tommasi, M., “Tree automata techniques and applications.” Available on: <http://www.grappa.univ-lille3.fr/tata>, 1997. release October, 1st 2002.