

# **CVPRIP - Calidad de Video Percibida en Redes IP**

Germán Bardier Valletta  
Carlos Fournier Soca  
Gastón Guridi Fernández

Tutor: José Joskowicz

Facultad de Ingeniería  
Universidad de la República

30 de Abril de 2009

Proyecto de fin de carrera, Ingeniería Eléctrica, Plan 97, Telecomunicaciones

## Agradecimientos

Este proyecto fue posible gracias al apoyo y contención de muchas personas.

A nuestras familias queremos agradecer por su apoyo incondicional, en especial a nuestras madres, padres y hermanos, además de su cariño hemos recibido de ellos albergue, transporte y comida, que no es poca cosa.

A Cecilia, Natalia y Yoko por su paciencia, amor y algún que otro reto.

A nuestro tutor José Joskowicz, agradecerle su ayuda y disposición.

A nuestros amigos, los mismos de siempre, a nuestros compañeros de trabajo, y en especial a todos los que nos ayudaron a sacar adelante la muy interesante tarea de observar muy divertidos videos, Nati, Tamy, Nando, Fermín, Gonchi, Carolina, Nico, el quinto elemento, Tamara, Coti, Victor, Nelson, Chispita, Andrés, Ximena, Rodrigo, Pino, Chuno, Chucky, Diego, Fernando, Florencia, Brunito, Cecilia, Mariana, Isa, Mauri, Cono y Elotamendi.

Queremos agradecer también a los colegas que desde distintas partes del mundo nos han dado una mano. Ellos son Amanda Alsafi del Institute for Telecommunication Sciences (ITS), Francisco Lopes y Fernando Ituo Higashie del Centro de Pesquisa e Desenvolvimento em Telecomunicações (CPqD), Filippo Speranza y Arthur Webster del Video Quality Experts Group (VQEG), Chulhee Lee de Yonsei University, y finalmente a Ignacio Irigaray, Pedro Casas, Gabriel Gomez, Eduardo Cota y Pablo Rodríguez Bocca de la Universidad de la República.

A la empresa Ingener S.A. queremos agradecer su desinteresada contribución.

Pedimos disculpas a todos aquellos que seguramente olvidamos mencionar.

**Muchas gracias..... Gastón, Germán y Carlos**

## Contenido

<b>Capítulo 1. Introducción .....</b>	<b>7</b>
1.1. Motivación.....	7
1.2. Objetivos del Proyecto.....	8
1.3. Estructura de la documentación .....	9
1.4. Tareas Realizadas .....	10
<b>Capítulo 2. Conocimientos previos y estado del arte .....</b>	<b>12</b>
2.1. Aspectos básicos del sistema visual humano .....	12
2.1.1. Anatomía .....	12
2.1.2. La Retina .....	13
2.1.3. Adaptación y enmascaramiento.....	15
2.2. Codificación de Video .....	16
2.3. Streaming de Video .....	19
2.4. Calidad perceptual de video.....	21
2.5. Resumen del VQEG .....	25
2.5.1. Fase 1 .....	25
2.5.2. Fase 2 .....	26
<b>Capítulo 3. Descripción del estudio .....</b>	<b>28</b>
3.1. Obtención de videos de referencia.....	28
3.2. Diseño del experimento.....	29
3.2.1. Codec.....	29
3.2.2. Parámetros de Codificación .....	30
3.2.3. Porcentaje y modelo de pérdida de paquetes .....	32
3.2.4. Selección de secuencias degradadas.....	33
3.3. Generación de videos degradados.....	35
3.3.1. Conversión de videos en formato .yuv a formato .avi.....	36
3.3.2. Codificación/Decodificación .....	36
3.3.3. Streaming de video/recepción del streaming.....	37

3.3.4. Perdedor de paquetes .....	38
3.3.5. Conversión de Formato sin Pérdida.....	40
<b>3.4. Descripción de las pruebas subjetivas .....</b>	<b>41</b>
3.4.1. Montaje del ambiente de prueba .....	41
3.4.2. Armado de las secuencias de video .....	43
3.4.3. Dimensionamiento de las pruebas .....	45
3.4.4. Observadores.....	47
<b>3.5. Descripción de los algoritmos para pruebas objetivas.....</b>	<b>50</b>
3.5.1. CPqD-IES .....	50
3.5.2. NTIA.....	50
3.5.3. PSNR .....	52
3.5.4. Desarrollo del algoritmo de Yonsei.....	53
3.5.4.1. Algoritmo.....	53
3.5.4.2. Post Ajustes.....	55
3.5.4.3. Diagrama del algoritmo.....	57
3.5.4.4. Validación del algoritmo.....	58
<b>3.6. Análisis estadístico .....</b>	<b>60</b>
3.6.1. Normalización de datos subjetivos. ....	60
3.6.2. Eliminación de sujetos.....	60
3.6.3. Ajuste de los datos.....	61
3.6.4. Métricas para la evaluación de los algoritmos.....	61
<b>Capítulo 4. Resultados obtenidos.....</b>	<b>67</b>
<b>Capítulo 5. Conclusiones .....</b>	<b>79</b>
<b>Capítulo 6. Bibliografía .....</b>	<b>81</b>
<b>Anexo A. Código fuente del algoritmo avi2yuv.....</b>	<b>84</b>
<b>Anexo B. Composición de las secuencias de video .....</b>	<b>86</b>
<b>Anexo C. Código fuente del algoritmo de Yonsei.....</b>	<b>90</b>
<b>Anexo D. Código fuente para el cálculo de las Métricas.....</b>	<b>94</b>
<b>Anexo E. Tablas de resultados de los test subjetivos.....</b>	<b>99</b>

<b>Anexo F. Tablas de resultados de los test objetivos .....</b>	<b>103</b>
<b>Anexo G. Análisis de la influencia de la pérdida de cuadros.....</b>	<b>105</b>
<b>Anexo H. Contenido del CD .....</b>	<b>107</b>

## Índice de Tablas

<b>Tabla 1.</b> Secuencias de referencia.....	28
<b>Tabla 2.</b> Selección de secuencias degradadas.....	34
<b>Tabla 3.</b> Características del monitor.....	42
<b>Tabla 4.</b> Valores de luminancia relevados.....	43
<b>Tabla 5.</b> Sujetos necesarios para el formato NTSC según los datos obtenidos del FRTV fase I .....	46
<b>Tabla 6.</b> Sujetos necesarios para el formato PAL según los datos obtenidos del FRTV fase I..	46
<b>Tabla 7.</b> Sujetos necesarios para el formato NTSC para este estudio.....	47
<b>Tabla 8.</b> Sujetos necesarios para el formato PAL para este estudio .....	47
<b>Tabla 9.</b> Resultados de las métricas para el formato PAL.....	72
<b>Tabla 10.</b> Resultados de las métricas para el formato NTSC .....	73
<b>Tabla 11.</b> Comparación de la varianza residual de los distintos algoritmos para el formato PAL .....	74
<b>Tabla 12.</b> Comparación de la varianza residual de los distintos algoritmos para el formato NTSC.....	74

## Índice de Figuras

<b>Figura 1.</b> Diagrama ilustrativo de las etapas de este trabajo .....	10
<b>Figura 2.</b> Diagrama del ojo humano.....	12
<b>Figura 3.</b> Estructura nerviosa de la retina.....	13
<b>Figura 4.</b> Distribución de fotorreceptores en la retina .....	14
<b>Figura 5.</b> Sensibilidad de los distintos tipos de fotorreceptores frente a la luz de distinta longitud de onda .....	15
<b>Figura 6.</b> Descomposición de una imagen en los componentes YCbCr.....	16
<b>Figura 7.</b> Valores de la componente Y, luego de aplicarse la transformada DCT y descartarse los valores con menor peso.....	17
<b>Figura 8.</b> Valores originales de la componente Y .....	17
<b>Figura 9.</b> Protocolos más utilizados discriminados según el modelo de capas TCP/IP .....	19
<b>Figura 10.</b> Esquema de encapsulamiento a través de las capas del modelo TCP/IP .....	20
<b>Figura 11.</b> Arquitectura cliente/servidor.....	21
<b>Figura 12.</b> Arquitectura P2P .....	21
<b>Figura 13.</b> Referencia completa, evaluación de la calidad perceptual.....	23

<b>Figura 14.</b> Referencia reducida, evaluación de la calidad perceptual en el receptor .....	23
<b>Figura 15.</b> Referencia reducida, evaluación de la calidad perceptual en el transmisor.....	24
<b>Figura 16.</b> Referencia Nula, evaluación de la calidad perceptual.....	24
<b>Figura 17.</b> Variación del DMOS en función de la tasa de bits de codificación .....	31
<b>Figura 18.</b> Efecto del entrelazado .....	32
<b>Figura 19.</b> Etapas del proceso de degradación .....	35
<b>Figura 20.</b> Vista de la herramienta ProCoder3 .....	37
<b>Figura 21.</b> Vista de la herramienta VLC .....	38
<b>Figura 22.</b> Vista de la herramienta Shunra.....	39
<b>Figura 23.</b> Arquitectura del laboratorio para la degradación de los videos .....	39
<b>Figura 24.</b> Vista frontal 1 .....	41
<b>Figura 25.</b> Vista frontal 2 .....	41
<b>Figura 26.</b> Vista general del ambiente de pruebas .....	42
<b>Figura 27.</b> Vista frontal del monitor .....	42
<b>Figura 28.</b> Vista del luxómetro .....	42
<b>Figura 29.</b> Composición de las secuencias de videos .....	43
<b>Figura 30.</b> Vista de la herramienta Vegas Movie Studio.....	44
<b>Figura 31.</b> Visualización del armado de las secuencias con la herramienta Vegas Movie Studio .....	45
<b>Figura 32.</b> Escala para la calificación de videos .....	48
<b>Figura 33.</b> Pasos requeridos para el cálculo del VQM.....	51
<b>Figura 34.</b> Visualización de las distintas etapas del tratamiento para cada cuadro en el algoritmo de Yonsei .....	54
<b>Figura 35.</b> Diagrama del algoritmo de Yonsei .....	57
<b>Figura 36.</b> VQM - Formato NTSC normalizado .....	58
<b>Figura 37.</b> VQM – Formato PAL normalizado.....	59
<b>Figura 38.</b> Diagrama de errores de clasificación .....	65
<b>Figura 39.</b> Calificación de los cuadrantes.....	65
<b>Figura 40.</b> Dispersión del DMOS y el VQM obtenido del algoritmo del CPqD para el formato PAL .....	68
<b>Figura 41.</b> Dispersión del DMOS y el VQM obtenido del algoritmo de la NTIA para el formato PAL .....	68
<b>Figura 42.</b> Dispersión del DMOS y el VQM obtenido del algoritmo de Yonsei para el formato PAL .....	69
<b>Figura 43.</b> Dispersión del DMOS y el VQM obtenido del PSNR para el formato PAL.....	69
<b>Figura 44.</b> Dispersión del DMOS y el VQM obtenido del algoritmo del CPqD para el formato NTSC.....	70
<b>Figura 45.</b> Dispersión del DMOS y el VQM obtenido del algoritmo de la NTIA para el formato NTSC.....	70
<b>Figura 46.</b> Dispersión del DMOS y el VQM obtenido del algoritmo de Yonsei para el formato NTSC.....	71
<b>Figura 47.</b> Dispersión del DMOS y el VQM obtenido del PSNR para el formato NTSC.....	71

<b>Figura 48.</b> Dispersión del DMOS y el DMOSp obtenido del algoritmo del CPqD para el formato PAL .....	75
<b>Figura 49.</b> Dispersión del DMOS y el DMOSp obtenido del algoritmo de la NTIA para el formato PAL .....	75
<b>Figura 50.</b> Dispersión del DMOS y el DMOSp obtenido del algoritmo de Yonsei para el formato PAL .....	76
<b>Figura 51.</b> Dispersión del DMOS y el DMOSp obtenido del PSNR para el formato PAL .....	76
<b>Figura 52.</b> Dispersión del DMOS y el DMOSp obtenido del algoritmo del CPqD para el formato NTSC.....	77
<b>Figura 53.</b> Dispersión del DMOS y el DMOSp obtenido del algoritmo de la NTIA para el formato NTSC.....	77
<b>Figura 54.</b> Dispersión del DMOS y el DMOSp obtenido del algoritmo de Yonsei para el formato NTSC.....	78
<b>Figura 55.</b> Dispersión del DMOS y el DMOSp obtenido del PSNR para el formato NTSC .....	78

## Capítulo 1. Introducción

En este capítulo se presenta los motivos que impulsan este estudio y sus objetivos. También se presenta la organización del documento y un breve resumen de las tareas realizadas a lo largo de todo el estudio.

### 1.1. Motivación

En el presente, Internet se ha impuesto como la red pública multiservicio de mayor difusión mundial, sobre la cual convergen mensajería, transmisión de datos, transmisión de voz, transmisión de video, entre otros.

Como es sabido, Internet es una red basada en IP cuya forma de trabajar es la de "best effort", no brindando garantías de ningún tipo sobre el servicio (QoS, priorización de paquetes, delay, jitter, etc). La red simplemente intentará transportar de la mejor manera los paquetes desde su origen a su destino.

Siendo para los usuarios de Internet imposible modificar los atributos de la misma más allá de su conexión, es imprescindible conocer como las características de la red afectan a las distintas aplicaciones. Esto toma una relevancia mayor si la aplicación resulta ser un servicio que se quiere vender.

Si entonces queremos transmitir video, se debe saber cómo las características de la red afectan a la calidad de éste servicio.

Para poder establecer esta relación, primero se debe tener claro cómo medir la calidad para cada servicio. En el caso de la transmisión/recepción de video, se ha aprendido con el tiempo que medir la calidad a través de métodos simples como el error cuadrático medio ente lo transmitido y lo recepcionado no refleja lo que al fin importa, lo que percibe el usuario del servicio.

Es aquí que entra un concepto clave, la calidad percibida es sin duda la llave que permite definir con precisión si un servicio de transmisión de video tiene una calidad acorde a la trazada o comprometida.

Teniendo en claro qué es lo que se debe medir (calidad percibida de video) surge necesidad de medir. Es en este punto que aparece la dificultad más grande. Para poder medir la calidad percibida, se necesita que las personas que reciben el video, transmitan su percepción. Parece claro que si es un servicio pago no es una forma viable de asegurar la calidad.

Se debe entonces, estimar de alguna manera la calidad percibida por los usuarios del servicio en el destino y poder monitorearla a través del tiempo, todo esto en forma automática.

En este punto se han realizado diversos trabajos por distintos laboratorios de empresas y universidades en todo el mundo, diseñando algoritmos que por distintos métodos y con distintas entradas estiman la calidad de video percibida.

Este trabajo se centra en los algoritmos surgidos de los trabajos del Video Quality Experts Goup (VQEG) y que luego fueron recomendados por la ITU (Internacional Telecommunication Union) resultando en la norma ITU-R BT.1683 [1], los cuales estiman la calidad percibida de video, teniendo como entradas la secuencia de video transmitida y la recepcionada (original y degradada).

Estos algoritmos se desempeñan correctamente cuando los videos se degradan por el proceso de codificación. Es claro que en un escenario en el cual

los videos son transmitidos existen otras fuentes de degradación. Según las condiciones de la red, esta última puede aportar a la degradación, por ejemplo a través de una posible pérdida de paquetes.

En resumen, este trabajo estudia el desempeño de los algoritmos antes referenciados en un escenario de transmisión de videos sobre redes IP, para satisfacer la necesidad de estimar la calidad percibida en el destino, y así poder comprometer niveles de servicios que permitan comercializar los mismos.

## 1.2. Objetivos del Proyecto

En la actualidad, los métodos utilizados para medir la calidad percibida son métodos subjetivos. Estos presentan varios inconvenientes, como ser intrusivos, requerir de la colaboración de terceras personas que realicen los test, no ser prácticos a la hora de un monitoreo en línea, etc.

Los ya referenciados estudios llevados adelante por el VQEG, proponen algoritmos para estimar la calidad percibida al observar un video degradado bajo ciertas condiciones, entre las que se encuentran la tasa de bits por segundo, el codec en que están codificados y la resolución.

Para poder simular la degradación de un video en una transmisión, se debe también tener en cuenta la degradación producida en la red. En las redes IP, esta degradación se puede representar por el porcentaje de pérdida de paquetes de la red (se mostrará más adelante en este trabajo).

**El objetivo de este trabajo es mostrar la performance de los algoritmos propuestos por la *Nacional Telecommunications and Information Administration* (NTIA – E.E.U.U.), el *Centro de Pesquisa e Desenvolvimento* (CPqD - Brasil) y la Universidad de *Yonsei* (Korea) surgidos de la fase II del VQEG, bajo la hipótesis de una degradación de video producida por una posible pérdida de paquetes en una transmisión sobre una red IP, junto con la degradación surgida del proceso de codificación. Este estudio se realizó para videos en formato PAL (625 líneas) y NTSC (525 líneas), es decir, televisión estándar.**

Las conclusiones a las que se ha arribado se sustentan mediante pruebas sobre el comportamiento de los algoritmos antes nombrados bajo las nuevas hipótesis, contrastando los resultados obtenidos contra el resultado de pruebas subjetivas llevadas adelante según las recomendaciones correspondientes.

### 1.3. Estructura de la documentación

Este documento se distribuye de la siguiente manera:

En el Capítulo 2 se recorren conceptos y conocimientos claves para la mejor comprensión de este estudio, como ser el sistema visual humano, codificación de video, streaming de video. Por otra parte se entrega un breve resumen del estado del arte en el estudio de la calidad percibida de video y de los aportes realizados por el *Video Quality Experts Group* a este campo.

El Capítulo 3 describe el estudio, desde las tareas previas necesarias para la realización del mismo como ser el diseño del experimento, la obtención de secuencias de referencia, el procesamiento de las secuencias de referencia, hasta la ejecución de las pruebas tanto objetivas como subjetivas, el procesamiento de los datos obtenidos.

Los resultados surgidos a partir del procesamiento de los datos provenientes de las pruebas se presentan en el Capítulo 4.

Finalmente en el Capítulo 5 se presentan las conclusiones finales del estudio.

## 1.4. Tareas Realizadas

Este proyecto fue llevado adelante en 5 etapas bien definidas. En la *Figura 1*, se muestran estas etapas y el orden en que fueron realizadas.

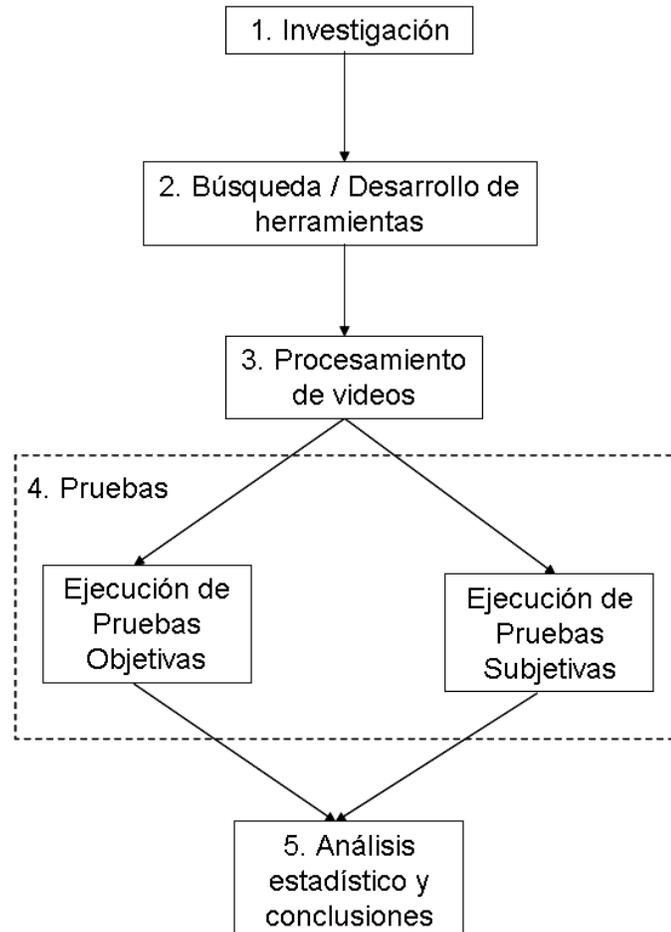


Figura 1. Diagrama ilustrativo de las etapas de este trabajo

Cada una de estas etapas implicó una serie de tareas, algunas de las cuales fueron ejecutadas en paralelo, y otras por su importancia fueron ejecutadas en forma individual.

A continuación y a modo de resumen se describen cada una de las etapas. Más adelante en este documento se detallan las tareas llevadas adelante.

- **Investigación:** En esta etapa se estudió el estado del arte en materia de calidad percibida de video. En este sentido se debió profundizar en conocimientos previos como el funcionamiento del sistema visual humano, así como lo referente a la codificación de video (YUV, AVI, MPEG, etc.), protocolos para la transmisión del video (RTP, RTSP, UDP, etc.). Posteriormente se recorren trabajos previos llevados adelante por distintos laboratorios del mundo, incluyendo proyectos de la Universidad de la República.

- **Búsqueda / Desarrollo de herramientas:** Un vez recorrida la etapa de investigación, se realizó una búsqueda distintos componentes claves para ejecutar el estudio. En esta búsqueda se encontraron herramientas ya desarrolladas y probadas para ejecutar distintas tareas, como ser la codificación de video o el streaming del mismo. Sin embargo, se debió desarrollar algunos programas para realizar tareas específicas como el cambio de algunos formatos de video. En lo que refiere a los algoritmos recomendados en la norma ITU-R BT.1683, se obtuvieron dos de ellos implementados por los laboratorios que los desarrollaron (NTIA y CPqD) mientras que se implementó el algoritmo desarrollado por la Universidad de Yonsei.
- **Procesamiento de videos:** Con las herramientas a disposición, se procedió a procesar los videos de referencia. Este procesamiento consistió en la codificación, transmisión y degradación por pérdida de paquetes. Previamente a la ejecución de este procesamiento se definió como procesar cada video, es decir, codec, tasas de compresión y porcentaje de pérdida de paquetes a aplicar a cada video.
- **Pruebas:** Contando con los videos procesados, se procedió a la ejecución tanto de las pruebas objetivas como de las pruebas subjetivas. Para éstas últimas se montó un ambiente de prueba acorde a las normas correspondientes.
- **Análisis estadístico y conclusiones:** Esta etapa comprendió el análisis completo de los resultados obtenidos de las pruebas objetivas y subjetivas. Se analizó como el resultado de los algoritmos se ajustan a las medidas subjetivas y cuál lo hace mejor en un escenario donde los videos se degradan por efecto de la pérdida de paquetes en una transmisión sobre redes IP, arribando a conclusiones sobre la validez de los algoritmos en dichas condiciones de estudio.

## Capítulo 2. Conocimientos previos y estado del arte

Distintas áreas de estudio atraviesan transversalmente a este estudio, y distintos conocimientos aplicados fundamentales para poder tener una visión acabada y poder llegar a conclusiones que no sólo sean sustentadas por resultados objetivos y válidos, sino que éstas tengan una aplicabilidad real para dar solución (o aportar en pro de ella) a problemas que hasta hoy no están resueltos.

Este capítulo recorre los distintos conocimientos y conceptos previos que resultan necesarios conocer a la hora de enfrentarse a un estudio de las características de éste. Estos temas se presentan en forma resumida ya que no es el objetivo de este trabajo profundizar desarrollar estos temas.

### 2.1. Aspectos básicos del sistema visual humano

Sin duda, conocer y comprender como funciona el sistema visual humano es clave para poder entender cómo se perciben las distintas degradaciones de una secuencia de video.

A continuación se ilustra cómo está compuesto el sistema visual humano y cómo funciona el mismo. Este trabajo no pretende explorar en profundidad este sistema, pero si proveer de nociones previas básicas sobre el mismo.

#### 2.1.1. Anatomía

En la *Figura 2* tomada del sitio web [2], se pueden apreciar las partes más importantes del ojo humano. No es parte de este estudio detallar la función de cada una de ellas por lo que solo se mencionaran las más relevantes.

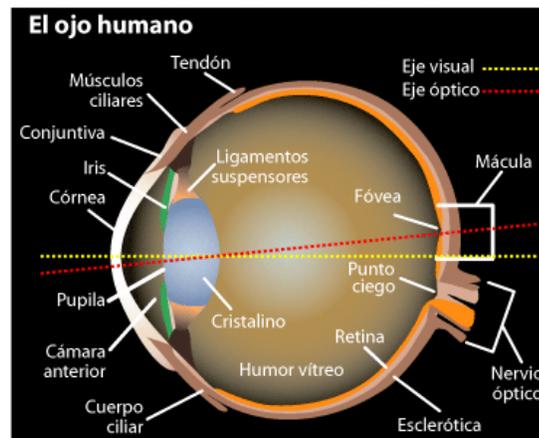


Figura 2. Diagrama del ojo humano

La luz que entra al ojo debe pasar a través de la cornea, el humor acuoso, el cristalino y por último el humor vítreo antes de llegar a la retina, que es la parte que realmente capta la imagen. Estas partes tienen la capacidad de variar el índice de refracción de forma de variar el ángulo de los rayos de luz incidentes lo que permite enfocar objetos a distintas distancias. Esta capacidad está dada por los músculos que controlan la curvatura del cristalino, cambiando su índice de refracción. Como el índice de refracción varía con la longitud de onda de la luz

incidente, no todos los colores pueden ser enfocados al mismo tiempo, esta característica se conoce como aberración cromática. La pérdida de la capacidad de enfocar objetos es una condición común en las personas de edad avanzada y lleva el nombre de presbicia.

La cantidad de luz que entra al ojo está regulada por la pupila, un grupo de músculos que pueden variar el diámetro de apertura entre 1.5 y 8 mm, lo que corresponde a una variación de hasta 30 veces en la cantidad de luz que entra al ojo [3].

### 2.1.2. La Retina

Como ya fue mencionado, la retina es la parte del ojo que capta las imágenes. Como se aprecia en la *Figura 3*, está formada por varias capas de neuronas y dos tipos de células fotorreceptoras, que cumplen funciones diferentes, los bastones y los conos. Aun no se sabe porque razón las células fotorreceptoras están ubicadas detrás de las neuronas en la retina. Los bastones son altamente sensibles a la cantidad de luz y por lo tanto son los que nos dan la capacidad de ver aun en ambientes con poca luz. Los conos son los responsables por la visión en colores, existen 3 tipos de bastones los cuales tienen su pico de sensibilidad a distintas longitudes de onda.

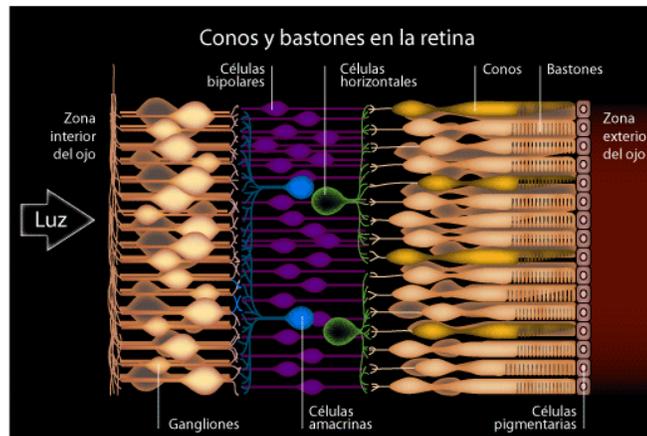


Figura 3. Estructura nerviosa de la retina

Las células horizontales, bipolares, amacrinas y gangliones procesan las señales recibidas por los conos y bastones, pueden tener salidas tanto inhibitoras como excitadoras. Los axones de los gangliones forman el nervio óptico que lleva las señales de la retina hasta el cerebro.

La luz de un solo fotón puede ser detectada por los bastones los cuales logran esta gran sensibilidad debido a que varios bastones actúan sobre una misma neurona, esta misma característica hace que se pierda resolución [3]. Esto hace que la visión en ambientes de poca luz no sea buena. La *Figura 4*, tomada de [2], muestra la distribución de conos y bastones en la retina.

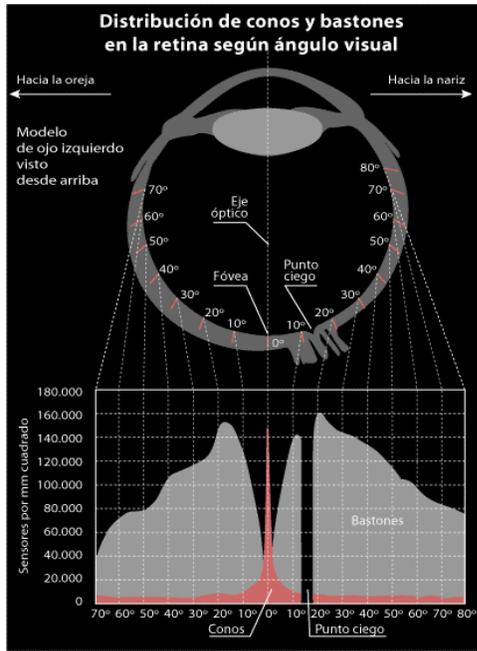


Figura 4. Distribución de fotorreceptores en la retina

Como se puede apreciar, la distribución de las células fotosensibles no es homogénea a lo largo de la retina. La zona que presenta la mayor densidad de conos es la fóvea, en esta zona se produce la visión detallada de los objetos. La densidad de bastones es mayor cuando estamos levemente desviados de la fóvea, esto explica el hecho que objetos de poca luminosidad, como puede ser una estrella, se perciban mejor si no centramos la vista en ellos. También se observa que en el punto ciego, lugar por donde el nervio óptico sale del ojo, carece completamente de fotorreceptores.

Los tres tipos de conos existentes reciben el nombre de conos S, conos M y conos L<sup>1</sup>. La cantidad de bastones supera a la de los conos en una relación 20 a 1, estando estos últimos concentrados en la fóvea. A su vez, la distribución de los conos tampoco es homogénea, con una proporción de 40 conos L y 20 conos M por cada cono S, aunque estos valores varían en cada individuo. Esta diferencia en la cantidad de cada tipo de conos genera una diferencia no despreciable en la resolución que se tiene para los conos S, lo cual es consistente con la menor exactitud de las longitudes de ondas más bajas, provocada por la aberración cromática [3]. En la *Figura 5*, se puede observar la sensibilidad de cada tipo de conos a las diferentes longitudes de onda. Se puede ver que la sensibilidad de los conos L y M es similar, mientras que la de los conos S se aleja sensiblemente. La superposición de los espectros es fundamental para poder diferenciar los distintos colores.

<sup>1</sup> Esta denominación proviene del inglés, dada por la longitud de onda a la que son sensibles cada tipo, Short (corta), Médium (media) y Long (larga).

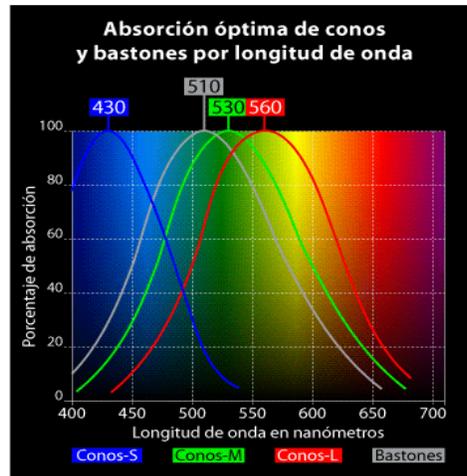


Figura 5. Sensibilidad de los distintos tipos de fotorreceptores frente a la luz de distinta longitud de onda

El área de la retina que influencia a una neurona tiene el nombre de campo receptivo, este concepto se extiende más allá de las neuronas de la retina y se puede hablar del campo receptivo en etapas posteriores del procesamiento visual. Los gangliones tienen campos receptivos de forma circular con una región central inhibitoria y una región externa excitatoria o viceversa. Cuando ambas regiones son estimuladas a la vez la respuesta es lenta, pero cuando la región excitatoria es estimulada y la inhibitoria no, la respuesta es rápida. Esta respuesta de las células ganglionares sirve para detectar bordes en las imágenes.

El sistema visual humano responde principalmente al contraste más que a la luminosidad. Si bien el umbral de detección de contraste no es constante para todos los niveles de luminosidad se aproxima bastante en un amplio rango, fuera de este rango nuestra sensibilidad al contraste disminuye, necesitando mayores niveles del mismo para detectar un estímulo. La sensibilidad al contraste también depende de la frecuencia temporal y espacial del mismo, y al igual que para la luminosidad, tiene un comportamiento pasa banda.

### 2.1.3. Adaptación y enmascaramiento

La adaptación y el enmascaramiento son dos fenómenos similares en los que la percepción de un estímulo se ve afectada por otro estímulo. En la adaptación, el sistema visual se adapta a un estímulo, reduciendo la sensibilidad al contraste frente a estímulos con similares características (orientación, frecuencia espacial, color). Si el sistema visual se adaptó a un estímulo de ciertas características y un nuevo estímulo tiene características similares será percibido con menor contraste.

El enmascaramiento se produce cuando un estímulo visible por sí mismo no puede ser percibido por la presencia de otro estímulo [3]. El enmascaramiento puede ser espacial o temporal, en el primero los estímulos son simultáneos, y en el segundo están separados temporalmente. A diferencia de la adaptación, que siempre reduce la sensibilidad, el enmascaramiento puede ayudar a percibir un estímulo en ciertos casos.

## 2.2. Codificación de Video

El objetivo de la codificación de video es transformar una señal de video analógico a video digital, de forma de poder almacenarlo, transmitirlo y procesarlo. Las características de la codificación dependerán del uso posterior que se le quiera dar al video, si es codificado para una aplicación de videoconferencia se buscará reducir el ancho de banda necesario mientras que codificado para DVD tendrá como principal objetivo mantener la calidad de la imagen. A pesar de esto, un fin común en la codificación de video es reducir la cantidad de memoria que ocupa. Para esto los distintos codificadores aprovechan las características del sistema visual humano de forma de eliminar información redundante o que no puede ser percibida. Esta información se encuentra tanto a nivel espacial como temporal y los distintos codecs (codificadores – decodificadores) utilizan diferentes técnicas para lograr estos beneficios. En la gran mayoría de los codecs el procesamiento en el codificador es mucho mayor que en el decodificador, esta asimetría tiene por objetivo tener decodificadores baratos que puedan ser masivos.

Los distintos tipos de archivo que usualmente se asocian a un video (.mpg, .avi, .flv, etc.) corresponden en realidad a un contenedor, es decir, una forma de guardar los datos dentro del archivo que no está necesariamente relacionada con el contenido del video o como está codificado este contenido. Típicamente, un video digital puede estar compuesto por uno o más flujos de información, cada uno con su respectiva codificación. Por ejemplo, 1 canal de video y 2 canales de audio, 1 canal de video sin canal de audio, etc. Es tarea de los contenedores manejar y sincronizar estos distintos flujos, así como guardar información como subtítulos, capítulos o metadatos.

Por otra parte están los codificadores propiamente dichos. Estos son los responsables por la compresión de los datos. Existen infinidad de codecs, cada uno con características que lo diferencian del resto y hacen su uso más o menos apropiado dependiendo de la finalidad. Algunos de los codecs más conocidos son MPEG-1 part 2, MPEG-2 part 2, WMV, DivX y MPEG-4 part 10 (H.264).

El primer paso con el que debe lidiar cualquier codec es la representación digital del color. El modelo tradicional, usado ampliamente en fotografía y video, es el RGB, a rojo-verde-azul, por sus siglas en inglés. Es un modelo de color aditivo en el que cualquier color puede ser representado por una combinación de los colores primarios, rojo, verde y azul. Si bien este espacio de colores resulta muy intuitivo, no es óptimo para el almacenamiento de la información, ya que posee mucha información redundante.

Para obtener el máximo provecho en la representación del color se suele utilizar el modelo YCbCr, el cual separa la información en la luminancia Y, y dos componentes de crominancia, Cb y Cr, tal como se ve en la *Figura 6*. Dado que la información de la luminancia es mucho más valiosa para el ojo humano, las crominancias suelen submuestrearse, dando lugar a formatos en los que puede haber 1 muestra de cada crominancia cada 2, 4 o hasta 8 muestras de luminancia [4].



Figura 6. Descomposición de una imagen en los componentes YCbCr

Existen técnicas que son comunes a la mayoría de los codecs, como son la utilización de la transformada discreta del coseno o los cuadros I, P y B, que explotan las redundancias espaciales y temporales respectivamente.

Buscando reducir la información necesaria para enviar un cuadro de video, se utiliza la transformada discreta del coseno (DCT, por sus siglas en inglés) aplicada sobre grupos de 8 x 8 pixeles, llamados comúnmente bloques. La DCT es una transformación similar a la de Fourier, pero solo utiliza valores reales. La aplicación de la DCT a los bloques transforma la información al dominio de la frecuencia, donde la mayoría de la información espacial puede ser representada por un número menor de valores. Una vez que se obtienen los coeficientes, estos son codificados utilizando codificación entrópica que permite reducir aún más la cantidad de bits necesarios para representar el cuadro de video. Esta técnica desarrollada por David A. Huffman, busca utilizar símbolos más cortos para los datos más frecuentes de forma de minimizar la cantidad de bits necesarios para representar los mismos datos. A diferencia de la transformación DCT, el proceso de codificación entrópica no conlleva pérdida de información [5].

$$\begin{bmatrix} 52 & 55 & 61 & 66 & 70 & 61 & 64 & 73 \\ 63 & 59 & 55 & 90 & 109 & 85 & 69 & 72 \\ 62 & 59 & 68 & 113 & 144 & 104 & 66 & 73 \\ 63 & 58 & 71 & 122 & 154 & 106 & 70 & 69 \\ 67 & 61 & 68 & 104 & 126 & 88 & 68 & 70 \\ 79 & 65 & 60 & 70 & 77 & 68 & 58 & 75 \\ 85 & 71 & 64 & 59 & 55 & 61 & 65 & 83 \\ 87 & 79 & 69 & 68 & 65 & 76 & 78 & 94 \end{bmatrix}$$

Figura 8. Valores originales de la componente Y

$$\begin{bmatrix} -26 & -3 & -6 & 2 & 2 & -1 & 0 & 0 \\ 0 & -2 & -4 & 1 & 1 & 0 & 0 & 0 \\ -3 & 1 & 5 & -1 & -1 & 0 & 0 & 0 \\ -4 & 1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Figura 7. Valores de la componente Y, luego de aplicarse la transformada DCT y descartarse los valores con menor peso

En la *Figura 8* se pueden ver los valores originales de uno de los componentes (Y, Cb o Cr) de un bloque. Una vez que se aplica la DCT y se descartan los valores con menor peso se obtiene una matriz resultante como la que vemos en la *Figura 7*, que al tener tantos valores nulos puede ser codificada ocupando menos memoria.

Normalmente, entre dos cuadros consecutivos muchos pixeles permanecen iguales y, dependiendo de la cantidad de movimiento del video, los que cambian son muchos menos. Resulta evidente que es más eficiente codificar solo los cambios entre un cuadro y el siguiente a codificar cada cuadro en su totalidad, incluyendo aquellos pixeles que permanece sin cambiar. Esta idea es la base de la codificación por compensación de movimiento. En esta técnica existen tres tipos de cuadros, los cuadros Intra (I), los cuadros predictivos (P) y los cuadros predictivos bidireccionales (B) [6]. Los cuadros I son codificados en su totalidad utilizando solamente técnicas de compresión intracuadro y por lo tanto son los cuadros de referencia. En los cuadros P se codifican solamente los cambios respecto al cuadro inmediatamente anterior de esta forma, conociendo el cuadro anterior y los cambios puede reconstruirse el cuadro P. Los cuadros B utilizan información tanto de cuadros anteriores como posteriores para su codificación, constituyendo los cuadros más eficientes en cuanto al espacio requerido.

Los distintos cuadros son luego agrupados formando lo que se conoce como grupo de imágenes (GOP, por sus siglas en inglés). Se entiende por GOP a la estructura de cuadros I, P y B entre dos cuadros I consecutivos. Por ejemplo, un

GOP puede ser de la forma IBBPBBPBBPBB. La distancia entre dos cuadros I consecutivos y la distancia entre dos cuadros P consecutivos son parámetros del codificador. El tamaño del GOP es de suma importancia para la eficiencia de la codificación, ya que los cuadros B y P ocupan mucha menos memoria que un cuadro I, por lo que un GOP mayor resulta más eficiente. Por otro lado, cuando hay errores o artefactos en el video, los cuadros B y P propagan estos errores a otros cuadros (cuando hay errores en los cuadros I o P ya que los Cuadros B no se utilizan para describir otros cuadros), llevando a que disminuya la calidad del video. En este sentido los GOP más cortos son más convenientes. Si bien los valores habituales en MPEG-2 son de 12 o 15 cuadros por GOP, no existe un límite máximo para el tamaño del mismo.

Otra técnica comúnmente utilizada es la compensación de movimiento. En muchos videos la única diferencia entre un cuadro y el siguiente es un movimiento en la cámara o en un objeto en particular, para tratar de aprovechar esta característica el codificador busca bloques de píxeles similares en cuadros anteriores o siguientes de forma de codificar solamente el vector de movimiento del bloque entre ambos cuadros. En particular, en MPEG-2, los bloques de píxeles que se buscan en la compensación de movimiento son de 16 x 16 píxeles y son buscados en una ventana de hasta 512 píxeles en cada dirección [7]. En codificadores actuales los bloques pueden ser de tamaño variable e incluso existe lo que se llama compensación global, en la que el codificador intenta detectar movimientos de toda la imagen, como puede ser producto del movimiento horizontal o vertical de la cámara. Otra posibilidad que ofrece la compensación de movimiento es detectar movimientos menores a un píxel (normalmente  $\frac{1}{2}$  o  $\frac{1}{4}$  píxel), en este caso, el codificador busca el movimiento interpolando píxeles contiguos. Un problema de la compensación de movimiento es la introducción de artefactos en la imagen, formado por los bordes de los bloques [8].

### 2.3. Streaming de Video

Originalmente para ver un video disponible a través de Internet se debía descargar completamente el archivo de video en su disco duro y luego reproducirlo con algún software local. Hoy en día, mediante la transmisión al instante de video (video streaming) se puede ver cualquier video mientras éste se descarga, directamente a través de un navegador. A fin de entender cómo se resuelve en la actualidad la transmisión de video, se muestra brevemente la concepción de una red IP, los principales factores y sus características.

Internet es una red IP basada en el modelo TCP/IP. Un modelo de red es una descripción a través de los protocolos y las tecnologías que la componen, estos se dividen en capas según sus funciones y aplicaciones. El modelo TCP/IP se describe en parte mediante los RFC 1122 [9], RFC 1123 [10], RFC 1349 [11]. Ver *Figura 9*.

Capa.	Algunos de los protocolos y estándares más usados.
5 Aplicación	HTTP <i>HyperText Transfer Protocol</i> , FTP <i>File Transfer Protocol</i> , DNS <i>Domain Name System</i> , RTP <i>Real-time Transport Protocol</i> , RTCP <i>RTP Control Protocol</i> , RTSP <i>Real-Time Streaming Protocol</i> .
4 Transporte	TCP <i>Transmission Control Protocol</i> , UDP <i>User Datagram Protocol</i> , SCTP <i>Stream Control Transmission Protocol</i> .
3 Red	IP (IPv4 o IPv6) <i>Internet protocol</i> , ARP <i>Address Resolution Protocol</i> , RARP <i>Reverse Address Resolution Protocol</i> , ICMP <i>Internet Control Message Protocol</i> .
2 Enlace	Ethernet, Token Ring, PPP, HDLC, Frame Relay, RDSI, ATM, IEEE 802.11.
1 Física	Cable coaxial, par trenzado, fibra óptica.

**Figura 9. Protocolos más utilizados discriminados según el modelo de capas TCP/IP**

El modelo se llama TCP/IP por los dos protocolos más importantes en Internet, TCP en la capa de transporte e IP en la capa de red. Cabe aclarar que si bien existen dos modelos aceptados mundialmente, uno aceptado por ser el teóricamente más correcto, el modelo OSI y el otro el más aceptado por ser de hecho el más utilizado, el modelo TCP/IP. En la actualidad existen otros modelos definidos por diferentes autores. En este estudio se utiliza como referencia el modelo TCP/IP.

Dentro del conjunto que compone el modelo anteriormente mencionado los protocolos más utilizados para realizar streaming de audio y video son UDP, RTP, RTSP y RTCP. RTP se utiliza para transportar los datos de audio y video, RTCP y RTSP se utilizan para controlar el flujo de datos y la creación de sesiones de transmisión entre usuario y servidor.

Dichos protocolos no orientados a la conexión permiten una comunicación de flujo continuo de paquetes de datos, ya que al ocurrir un error de transmisión estos continúan enviando datos. En este tipo de comunicación es más importante mantener una comunicación fluida e ininterrumpida, que una sin errores. Por esta última razón es que los paquetes no se retransmiten, y que se mantiene un buffer pre cargado antes de empezar a reproducir. El buffer evita que el reproductor se quede sin datos en plena reproducción.

Los paquetes de datos son encapsulados utilizando el protocolo RTP en la capa de aplicación, para luego, en la capa de transporte, ser encapsulados en datagramas utilizando el protocolo UDP. Finalmente en la capa de red son encapsulados en paquetes, utilizando el protocolo IP. Se le llama encapsulado al procedimiento por el cual en cada capa, según el protocolo correspondiente, se agrega a la unidad de datos proveniente de la capa superior un encabezado y una cola.

Este procedimiento se realiza en orden inverso en el destino de la información. Un esquema del encapsulamiento a través de las distintas capas puede verse en la Figura 10.

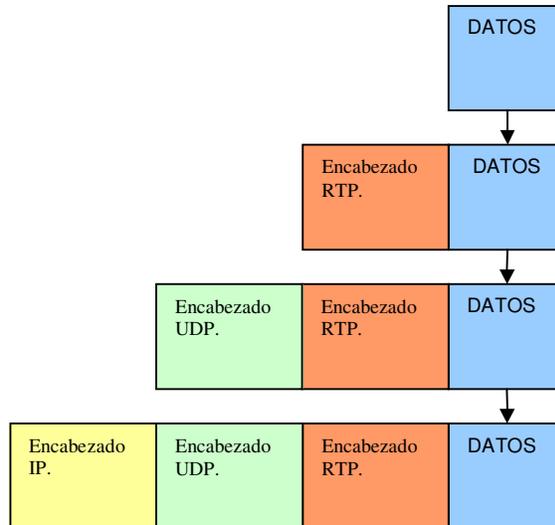


Figura 10. Esquema de encapsulamiento a través de las capas del modelo TCP/IP

Hoy son muchas las aplicaciones de Streaming en vivo bajo demanda, por ejemplo, transmisiones de radio y televisión, conciertos musicales, eventos deportivos, aplicaciones medicas, congresos y conferencias, educación a distancia, video CHAT, visualización remota de distintas actividades, presentaciones on line.

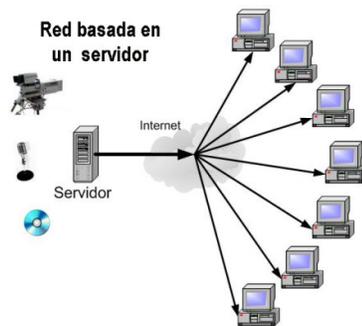
Como se puede ver en los diferentes tipos de aplicaciones que utilizan el streaming de video, este puede ser utilizado de usuario a usuario, o con estructuras del tipo cliente/servidor, o varios servidores y varios clientes. Esta última estructura es la más utilizada hoy en día ya que optimiza los tiempos de descarga y de utilización. Diferentes servidores distribuidos geográficamente con el mismo contenido permiten un tiempo online del 100%.

Actualmente, las limitantes que las redes IP presentan como las pérdidas de paquetes, demoras, jitter, ancho de banda limitado, etc, hacen que este medio no sea el más utilizado para entregar distintos servicios (por ejemplo la televisión para abonados). Sin embargo, cada vez más servicios utilizan esta tecnología y más usuarios consumen los mismos.

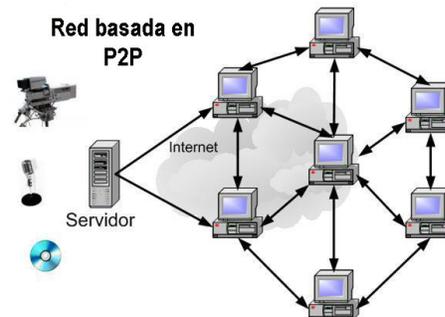
## 2.4. Calidad perceptual de video

Existen hoy en día varias técnicas para transmitir video en redes IP, entre las más difundidas se pueden mencionar la transmisión puerto a puerto (P2P - Peer to Peer), o la transmisión del tipo IPTV. La técnica P2P TV utiliza una arquitectura consistente en la distribución de un archivo en diferentes nodos y la redistribución de este al nodo que realizó el "request". El nodo central es la fuente del archivo y solo participa en el inicio de la transmisión.

La transmisión del tipo IPTV utiliza un nodo central, en general de un proveedor privado o un servidor público. Este servidor distribuye la señal directamente al cliente. En la *Figura 11* y la *Figura 12* se pueden observar ambas arquitecturas.



**Figura 11. Arquitectura cliente/servidor**



**Figura 12. Arquitectura P2P**

En lo que a servicios de telefonía se refiere, tanto la telefonía tradicional como los sistemas de telefonía celular, y los sistemas de telefonía IP, tienden a las video llamadas.

Es evidente que evaluar la calidad de los servicios de video en los diferentes sistemas es cada día más imprescindible, de forma de asegurar la calidad del servicio.

A la hora de calificar un sistema de video no existen mejores evaluadores que los propios usuarios. Por este motivo existen distintos métodos subjetivos utilizados para evaluar los sistemas de video en laboratorio. Estos métodos se basan en la observación por parte de personas de series de videos, los cuales son calificados según la calidad de cada uno. Los videos procesados son el resultado del sistema a evaluar, realizando las observaciones en ambientes controlados del tipo laboratorio.

Dichos métodos están normalizados a través de la ITU y son de uso aconsejable ya que están sumamente probados. En tal caso, simplemente debe elegir el método más adecuado para la aplicación a evaluar.

Existen diferentes enfoques para este tipo de evaluación, los hay de evaluación absoluta y los de evaluación relativa. Del primer tipo se puede mencionar el ACR (Absolute Category Rating) descrito en la norma ITU-T P.910 [12]. El método consiste en mostrar a los sujetos de prueba, una serie de videos procesados, uno después del otro, debiendo cada sujeto evaluar la calidad de cada video mediante una escala discreta del 1 al 5, siendo 1 equivalente a Malo y 5 equivalente a Excelente.

El test de evaluación relativa más reconocido por su independencia a los efectos contextuales es el Double-Stimulus Continuous Quality-Scale (DSCQS), descrito en la norma ITU-R BT.500-11 [13]. Este método consiste en presentar

pares de videos original-degradado a los observadores, debiendo ellos evaluar ambos videos, conociendo que uno es el original pero sin saber cual. El resultado de este método es un Valor Medio de Opinión Diferencial (DMOS por su sigla en inglés). El rango de este valor va de 0 a 1, donde 0 significa que la diferencia entre ambos videos es imperceptible al ojo humano y 1 equivale a que ambos videos son completamente diferentes.

Estas pruebas subjetivas se deben realizar en ambientes controlados del tipo laboratorio, los monitores deben estar calibrados mediante sistemas del tipo PLUGE, el ambiente de pruebas debe estar bien definido, la distancia y el ángulo de observación están especificados. La luminancia del ambiente de prueba debe ser medida y controlada. A su vez, los sujetos de pruebas se deben validar, es decir que se les debe realizar pruebas de visión, agudeza (Test Snellen), distinción del color (Test Ishihara) y percepción del contraste (Test Pelli-Robson).

Las pruebas subjetivas realizadas de forma correcta son el método más confiable y fiel para evaluar un sistema de video, obteniéndose valores de la calidad percibida que representan la opinión de la media poblacional (valores "reales"), pero presentan grandes inconvenientes.

El primer inconveniente, es que carecen de practicidad frente a los servicios online o de transmisión en vivo. Son varios los casos en los cuales es imposible medir la calidad de video percibida mediante cualquier método del tipo subjetivo. Algunos de estos ejemplos de medidas que no se pueden realizar mediante métodos subjetivos son:

1. Monitoreo de la calidad en llamada de video Usuario-Usuario.
2. Monitoreo de la recepción de IPTV o VoD (Video on Demand).
3. Estimación de la calidad de video recibida por los usuarios de un sistema de Multicast TV, mediante variables del canal de transmisión.
4. Predicción del deterioro de la calidad perceptual de video al ingresar nuevos usuarios o frente a cambios de variables en un sistema del tipo IPTV (Un solo servidor central).

En la mayoría de los casos existe un interés económico detrás de la medición de la calidad de los servicios, pues un usuario al adquirir el servicio exigirá cierta calidad de servicio (Quality of Service - QoS) y a su vez el proveedor deberá poder estimar si la venta de nuevos servicios, de un servicio a nuevos usuarios, deteriora o no el sistema general de forma de implementar mejoras al sistema.

Es de destacar que se estiman en más de 10 millones la cantidad de usuarios de servicios de IPTV según Point Topic [14]. La mayoría de las grandes cadenas televisivas están haciendo inversiones en el tema. Existen hoy en día una variedad de empresas que ofrecen este tipo de servicio, desde la BBC hasta empresas sin emisoras tradicionales de TV como Babelgum. También la telefonía IP viene reemplazando poco a poco a la telefonía tradicional, y aplicaciones como Skype, MSN live o NetMeeting, originalmente aplicaciones de chat y telefonía IP, ya prestan hoy servicios de video llamadas gratuitas.

Otro de los grandes inconvenientes de las pruebas subjetivas es el tiempo consumido en realizarlas. Para obtener resultados válidos se deben realizar pruebas con una cantidad de sujetos que asegure que los resultados representan la opinión de la media poblacional. Al tiempo necesario para que cada sujeto observe los videos se le debe agregar los tiempos de coordinación de las sesiones, y el tiempo dedicado a preparar el ambiente de pruebas, armar un pequeño escenario acondicionado con los equipos apropiados. Finalmente el tiempo de procesamiento y análisis estadístico que requiere este tipo de pruebas.

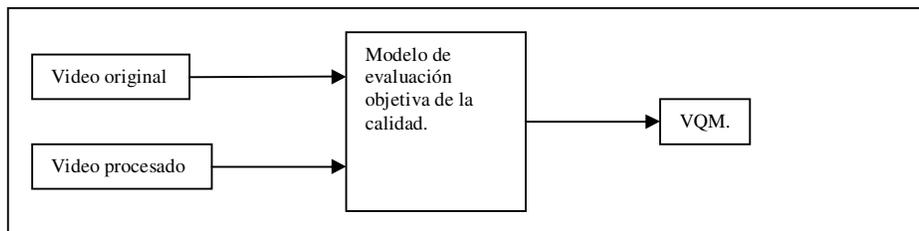
Si se toma como ejemplo las pruebas de la fase I o II del VQEG, todo el proceso de pruebas subjetivas tomó varios meses a un grupo de expertos internacionales en laboratorios y universidades de primer nivel.

Frente a la creciente demanda de aplicaciones de transmisión de video online y las ineficiencias de las pruebas subjetivas, es clara la necesidad de una herramienta objetiva, rápida y confiable para evaluar la calidad percibida de video en el destino.

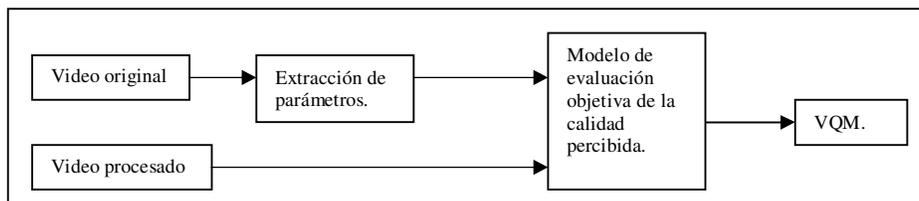
Dentro de este campo existen muchos estudios, un ejemplo de ello son los trabajos realizados por el VQEG a través de su proyecto FRTV ([15] y [16]), pero no son los únicos, de hecho otros laboratorios y universidades alrededor del mundo han presentado sus estudios sobre el tema (por ejemplo los trabajos realizados por Pastrana-Vidal & Gicquel en 2006 [17] o el realizado por Le Callet, Viard-Gaudin & Péchard en 2005 [18]). Se pueden distinguir tres tipos de sistemas, los sistemas de referencia completa (Full Reference - FR), los de referencia reducida (Reduced Reference - RR) y sin referencia alguna (Null Reference - NR). En los sistemas sin referencia (obviamente los más complejos), la idea es lograr una evaluación de la calidad de un video sin tener información en lo absoluto del video original. Esta sería la herramienta ideal, pero en la actualidad ninguna herramienta de este tipo alcanza una performance aceptable, de modo que se utilizan herramientas que evalúan la calidad perceptual mediante la comparación del video procesado y una referencia del video original.

La información del video original utilizada para la evaluación de la calidad, puede variar de una propuesta a otra, por ejemplo, se pueden observar distintos enfoques en los trabajos llevados adelante por Oelbaum & Diepold en 2007 [19], por Wolf & Pinson en 2005 [20] o por Le Callet, Viard-Gaudin & Péchard en 2005 [18]. Incluso se plantean sistemas que reenvían información de la calidad percibida al servidor, realizándose la evaluación de la calidad en el transmisor (ver figura 5). Siempre que se envía información adicional para realizar la evaluación de la calidad, ya sea del transmisor al receptor o de este último al transmisor, aconsejable utilizar un canal diferente y más seguro que el canal utilizado para enviar el video. Estos sistemas están pensados para canales de transmisión con pérdidas, si se utiliza el mismo canal para enviar las referencias de comparación y estas se pierden, la evaluación no sería buena.

En las figuras *Figura 13*, *Figura 14*, *Figura 15* y *Figura 16* se pueden observar las diferentes estructuras de los enfoques mencionados.



**Figura 13. Referencia completa, evaluación de la calidad perceptual**



**Figura 14. Referencia reducida, evaluación de la calidad perceptual en el receptor**

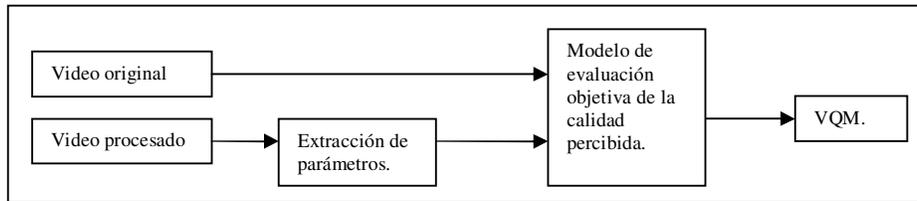


Figura 15. Referencia reducida, evaluación de la calidad perceptual en el transmisor

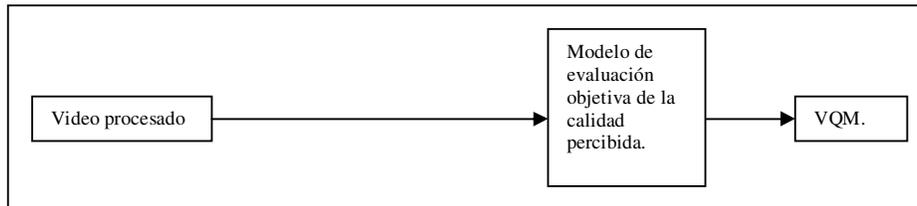


Figura 16. Referencia Nula, evaluación de la calidad perceptual

Los métodos Full Reference son por el momento los más utilizados. Dentro de esta categoría el primero a mencionar es el típico PSNR (Peak Signal to Noise Ratio). Para calcularlo se toma el video original y el video procesado, se calcula el error cuadrático cuadro a cuadro, luego se calcula la media de este error y sobre esto se computa el PSNR. Además del PSNR existen otras posibilidades, comenzando por los códigos normalizados en las normas ITU-T Rec-J.144 [21] e ITU-R Rec-BT.1683 [1], donde podemos encontrar un PSNR de bordes (Modelo de la Universidad de Yonsei), un modelo que detecta varios parámetros entre video original y degradado y luego los integra con pesos apropiados para obtener el VQM deseado (BT British Telecom), o un modelo basado en la segmentación de la imagen (CPqD - Telecommunications Research and Development Center). Dentro de las normas mencionadas anteriormente también se detalla el modelo propuesto por la NTIA.

Todos los modelos dentro de estas normas fueron reconocidos como estadísticamente equivalentes todos entre sí y significativamente superiores al DMOS estimado por el PSNR. Mientras que solo cuatro algoritmos están normalizados, se puede encontrar cada día un nuevo proponente. A continuación mencionamos otros trabajos donde se exponen nuevos modelos.

- *Objective quality assessment of color images based on a generic perceptual reduced reference*, propuesto por Mathieu Carnec, Patrick Le Calleta y Dominique Barbaa de la École Polytechnique de l'Université de Nantes [22]. Modelo de referencia completa basado en la descomposición de las muestras en sub-bandas en el espacio de la frecuencia.
- *On Evaluating Video Object Segmentation Quality: A Perceptually Driven Objective Metric*, propuesto por Elisa Drelie Gelasca y Touradj Ebrahimi [23]. Modelo basado en segmentación y caracterización de si los errores entre video original y procesado afectan o no la forma del objeto visualizado.
- *Objective Perceptual Video Quality Measurement using a Foveation-Based Reduced Reference Algorithm*, propuesto por Fang Meng, Xiuhua Jiang, Hui Sun y Shuang Yang [24]. Modelo de referencia reducida, basada en generación de capas de espacio-temporales en similares a las encontradas en la región fóvea del ojo humano.

Al día de hoy, las nuevas aplicaciones y la demanda de los consumidores frente a ellas hacen necesaria la búsqueda de formas de evaluar de manera objetiva la QoE (Calidad de la experiencia por sus siglas en inglés) o la QoS. Si bien las técnicas subjetivas nunca perderán su lugar, pasaran a un plano de referencia y no de medida directa. Aunque aún se está muy lejos de obtener una métrica lo suficientemente avanzada, que contemple el contenido de las imágenes como lo hace un ser humano, el camino se está recorriendo, desde los modelos de referencia completa y apuntando a los de referencia nula.

## 2.5. Resumen del VQEG

El Video Quality Experts Group (VQEG), es un grupo de expertos con diversa formación, de distintas y muy reconocidas organizaciones de todo el mundo, que trabajan en el campo de la evaluación de la calidad de video. El grupo fue formado en Octubre de 1997 en un encuentro de expertos en calidad de videos. La mayoría de los participantes están activos en la International Telecommunication Union (ITU). El VQEG combina los recursos y experiencia que se encuentran distribuidos en distintos grupos de estudio de la ITU, trabajando para alcanzar una meta común [25].

Este grupo llevó adelante, entre otros proyectos relacionados a la calidad de video, el proyecto FRTV, dedicado al estudio de la calidad de video para servicios de TV en Standard Definition (PAL-N y NTSC). El propósito de este proyecto consistió en estudiar como se ve afectada la calidad percibida por las personas, al degradar un video con la definición antes indicada mediante la codificación, utilizando distintos codecs a distintas tasas de compresión, e intentar obtener un modelo que permitiera obtener una estimación objetiva de esta degradación.

Este proyecto fue realizado en dos fases, cada una de las cuales arrojó distintos resultados. Los mismos se resumen a continuación:

### 2.5.1. Fase 1

Para esta primera etapa se analizó un amplio conjunto de datos. Diez participantes propusieron un modelo cada uno para calcular objetivamente la calidad percibida de video, comparando los resultados contra pruebas subjetivas realizadas sobre un amplio rango de secuencias de video en distintos sistemas de video. Más de 26.000 medidas subjetivas fueron recogidas sobre la base de veinte secuencias de video de referencia, procesadas en dieciséis sistemas de video distintos, en ocho laboratorios independientes alrededor del mundo [15].

Estas pruebas subjetivas se realizaron siguiendo estrictamente las pautas indicadas por la norma ITU-R BT.500-8 [26].

En resumen, siete u ocho de los diez modelos surgidos en esta fase son estadísticamente equivalentes entre sí (dependiendo de la métrica) y equivalentes al Peak Signal-to-Noise Ratio (PSNR).

Los resultados no permitieron al VQEG proponer ningún modelo objetivo para recomendar su inclusión en las normas ITU, pero permitió avanzar en el estado del arte en el campo de la medición objetiva de la calidad percibida de video.

Sin lugar a duda, la información generada en oportunidad de este proyecto significó mucho para los desarrolladores. En particular, previamente a este

proyecto no existía una biblioteca tan grande de secuencias degradadas sometidas a pruebas subjetivas.

### 2.5.2. Fase 2

La segunda fase se compuso de dos evaluaciones paralelas de videos de referencia. Una evaluación realizada por paneles de observadores y la otra por programas computacionales que modelan objetivamente la calidad de video. Estos modelos objetivos pretenden predecir los juicios subjetivos emitidos por los seres humanos [16].

En esta segunda fase se estudiaron dos definiciones de imagen, 525 líneas (NTSC) y 625 líneas (PAL-N). Cada prueba abarcó un amplio rango de calidades, así como secuencias de las más diversas desde el punto de vista de la cantidad de movimiento, detalle espacial, color, etc. De esta forma se evaluó el desempeño de cada uno de los modelos y sus diferencias estadísticas.

Los resultados fueron expuestos en términos del Differential Mean Opinion Score (DMOS) – medida cuantitativa de la calidad subjetiva de una secuencia de video juzgada por un panel de observadores. Los DMOS obtenidos de los test en NTSC se distribuyeron en un rango más amplio que en PAL-N (0 a 80 y 3 a 55 respectivamente en una escala de 0 a 100).

Tres laboratorios independientes llevaron adelante las pruebas subjetivas, Communications Research Center (CRC, Canadá) y Verizon (EEUU) realizaron las pruebas en NTSC mientras que la Fondazione Ugo Bordoni (FUB, Italia) realizó las pruebas en PAL-N. Todas estas pruebas subjetivas fueron realizadas en ambientes controlados, siguiendo como guía la norma ITU-R BT.500-10 [27], forma de asegurar que las mismas fueran válidas y comparables.

Inicialmente, diez laboratorios de todo el mundo se postularon para el desarrollo y pruebas de los algoritmos, pero solo seis de ellos completaron los test. Estos seis laboratorios son Chiba University (Japón), British Telecom (Reino Unido), CPqD (Brasil), NASA (EEUU), NTIA (EEUU), y Yonsei University / Radio Research Laboratory (Corea). Todos los laboratorios proponentes, produjeron algoritmos que modelan objetivamente la calidad de video, y sometieron a las mismas secuencias utilizadas en las pruebas subjetivas a los test objetivos.

Para comparar los distintos modelos, tanto con las pruebas subjetivas como entre sí, se sometieron los resultados a distintos test estadísticos que dieron lugar a distintas métricas.

- Métrica 1: Coeficiente de correlación lineal de Pearson ente el DMOSp (predicto por los modelos) y el DMOS surgido de las pruebas subjetivas.
- Métrica 2: Coeficiente de correlación de Spearman entre el DMOSp y el DMOS.
- Métrica 3: Cociente de Outliers
- Métricas 4, 5 y 6: Estas métricas fueron evaluadas según se describe en la norma ITU-R BT.1676 [28].
  - Métrica 4: RMS Error.
  - Métrica 5: Resolving Power.
  - Métrica 6: Classification Errors.
- Métrica 7: Test F del error residual de los modelos en comparación con el error residual de un “modelo óptimo”.

Los resultados obtenidos (tanto para NTSC como para PAL-N) son similares pero no idénticos.

De este estudio surgió en primera instancia la recomendación y luego la normalización por parte de la ITU de cuatro de los seis modelos desarrollados. Los algoritmos normalizados son los desarrollados por British Telecom (UK), CPqD (Brazil), NTIA (USA), y Yonsei University / Radio Research Laboratory (Korea). La norma que resulto es la ITU-R BT.1683 [1].

## Capítulo 3. Descripción del estudio

En este capítulo se describen detalladamente las distintas etapas del estudio, las definiciones tomadas y las herramientas utilizadas. También se exponen algunos problemas y obstáculos que se enfrentaron así como las alternativas halladas para superar los mismos.

### 3.1. Obtención de videos de referencia

Para este estudio, resulta clave que los videos de referencia recorran los distintos tipos de contenido con sus respectivas características como son el color, la cantidad de movimiento, etc. También, estas secuencias de referencia no deben estar degradadas desde ningún punto de vista.

Las secuencias utilizadas por el VQEG en la fase I [15] del proyecto FRTV cumplen con las características antes mencionadas. Por otra parte, algunas de estas secuencias fueron utilizadas por el VQEG para la fase II [16] del FRTV, lo que resulta extremadamente útil para comparar resultados, y en particular para evaluar la correcta implementación del algoritmo desarrollado por la Universidad de Yonsei (3.5.4.4).

El VQEG ha hecho públicas las secuencias utilizadas para VQEG en la fase I [15] en su sitio web [25], de donde han sido obtenidas para la realización de este estudio. A continuación se detallan las secuencias.

Secuencias		Características	Fuente
src1	Tree Still	Diferentes direcciones	EBU
src2	Barcelona	Color saturado + efecto de enmascaramiento	RAI/Retelevision
src3	Harp	Color saturado, acercamiento, realzado, detalles finos	CCETT
src4	Moving graphic	Critico para Betacam, color, texto en movimiento, detalles finos, sintético	RAI
src5	Canoa	Movimiento de agua, movimiento en diferentes direcciones, alto nivel de detalle	RAI
src6	F1 Car	Movimiento rápido, color saturado	RAI
src7	Fries Film	Texturas, paneo rápido	RAI
src8	Horizontal scrolling 2	Desplazamiento de texto	RAI
src9	Rugby	Movimiento y colores	RAI
src10	Mobile&calendar	Disponible en los dos formatos, color y movimiento	CCETT
src13	Baloon-pops	Película, color saturado, movimiento	CCETT
src14	NewYork 2	Efecto de enmascaramiento, movimiento	AT&T/CSELT
src15	Mobile&Calendar	Disponible en los dos formatos, color y movimiento	CCETT
src16	Betes_pas_betes	Color, sintético, movimiento, escenas cortadas	CRC/CBC
src17	Le_point	Color, transparencia, movimientos en todas las direcciones	CRC/CBC
src18	Autumn_leaves	Color, paisaje, acercamiento, movimiento de salto de agua	CRC/CBC
src19	Football	Color, movimiento	CRC/CBC
src20	Sailboat	Casi sin movimiento	EBU
src21	Susie	Texturas	EBU
src22	Tempete	Color, movimiento	EBU

Tabla 1. Secuencias de referencia

## 3.2. Diseño del experimento

El universo de variables que pueden encontrarse cuando de codificación y transmisión de video se habla son innumerables. En este trabajo se diseñó el experimento y sus variables de forma de cumplir con el objetivo del estudio sin apartarse de la realidad tecnológica del mercado en cuanto a transmisión de video en tiempo real en Internet se trata.

### 3.2.1. Codec

En este marco, la primera variable que se determinó fue el codec a utilizar para codificar los distintos videos de referencia, que como se mencionó anteriormente, no tienen ningún tipo de codificación o compresión. Es así que analizando el estado del arte y la realidad actual de los prestadores del servicio de transmisión de video sobre Internet se definió en un comienzo por utilizar dos codecs distintos. MPEG-2 y H.264 (MPEG-4 part 10 avc).

MPEG-2 es el estándar de la industria al día de hoy. “Es la designación para un grupo de estándares de codificación de audio y vídeo acordado por MPEG (por la sigla en Inglés de Grupo de Expertos en Imágenes en Movimiento), y publicados como estándar en la ISO 13818” [29].

Este codec es comúnmente utilizado para servicios de transmisión de audio y video como ser la televisión digital (por cable o satélite indistintamente). MPEG-2 es también el codec utilizado para la confección de los DVD's que se comercializan hoy en día.

El formato de codificación MPEG-2 es muy similar a su predecesor MPEG-1. La principal diferencia de interés desde el punto de vista de los servicios de transmisión de TV en definición estándar es que MPEG-2 soporta video entrelazado, formato utilizado por las televisiones.

Es a partir de 3 Mbit/s que MPEG-2 tiene un desempeño óptimo. Para tasas inferiores a 1 Mbit/s, MPEG-2 tiene una performance inferior a la de MPEG-1 [29].

H.264 o MPEG-4 parte 10 es el formato de codificación de video que se perfila como el nuevo estándar de la industria. H.264 “es una norma que define un códec de vídeo de alta compresión, desarrollada conjuntamente por el ITU-T Video Coding Experts Group (VCEG) y el ISO/IEC Moving Picture Experts Group (MPEG)” [30].

Por contraposición a MPEG-2, H.263 y MPEG-4 parte 2, H.264 tiene un buena performance a tasas bajas sin incrementar la complejidad de su diseño.

En H.264 la compresión determinada por la transformada DCT + Compensación de Movimiento de las versiones anteriores no se modificó por considerarse que era superior a otros estándares. Sin embargo otras tecnologías descartadas en el pasado por su complejidad y su alto costo de implementación se volvieron a examinar (por ejemplo, VLSI).

Una de las principales diferencias entre H.264 y MPEG-2 radica en la forma en que se compensa el movimiento. Cada macrobloque de 16x16 pixeles se puede descomponer en sub-bloques de 16x8, 8x16, o 8x8 pixeles. A su vez, estos últimos pueden ser descompuestos en particiones de 8x4, 4x8, o 4x4 [30].

A pesar de haberse definido en primera instancia la utilización de este codec, este formato de codificación fue descartado por motivos de implementación.

Habiéndose recorrido una importante variedad de sistemas de streaming de video, y distintas combinaciones cliente/servidor para implementar el streaming de video codificado en H.264, no se encontró ninguno que funcionara correctamente.

En particular, el Darwin Streaming Server modifica la cantidad de cuadros por segundo de los videos de referencia, lo cual es inaceptable para este estudio. Por otra parte, el cliente VLC no pudo recepcionar correctamente los flujos de H.264, obteniéndose en recepción secuencias entrecortadas y con artefactos. El sistema compuesto por el cliente y el servidor VLC funciona correctamente cuando se trabaja con un GOP sin cuadros B, pero una vez más, esto resulta inaceptable para cumplir con los requerimientos de este estudio.

Es importante destacar que en el mercado existen soluciones para el streaming de H.264, pero todas ellas implementadas en hardware y con valores comerciales que escapan totalmente el presupuesto con el cual se ejecutó este estudio.

### 3.2.2. Parámetros de Codificación

Una vez seleccionado MPEG-2 como codec, es necesario definir los parámetros de codificación como ser la tasa de bits, GOP, etc.

En primera instancia, se definió las tasas de bits a las cuales las secuencias de video fueron codificadas a la postre.

La *Figura 17*, extraída de “*A Mathematical Model for Evaluating the Perceptual Quality of Video*” [31], muestra la “variación de la calidad percibida (medido como el DMOS, normalizado entre 0 y 1) en función de la tasa de bits, manteniendo constante los demás parámetros, para la secuencia “Football” (src 19), codificada en MPEG-2. La figura muestra el comportamiento típico de cualquier otra secuencia en estas condiciones” [31].

En este ambiente de degradación únicamente causada por la codificación (sin pérdida de paquetes), la relación entre la tasa de bits y el DMOS tiene un comportamiento casi plano para tasas superiores a los 6 Mbist/s, y un DMOS muy malo para tasas inferiores a 900 Kbits/s. Por tanto, se definió codificar las secuencias a tasas entre 900 Kbits/s y 6 Mbits/s, concentrando más puntos a tasas bajas, ya que para tasas altas el comportamiento tiende a ser plano. Específicamente, las tasas a las cuales se definió codificar fueron: 900 Kbits/s, 1.2 Mbits/s, 1.5 Mbits/s, 2.0 Mbits/s, 3.0 Mbits/s y 6.0 Mbits/s.

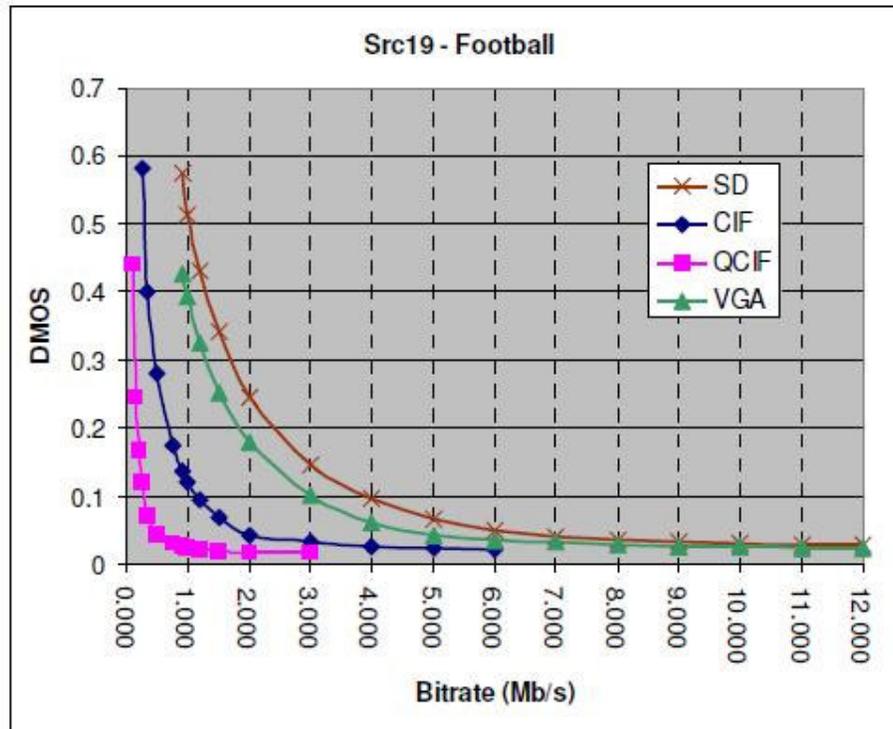


Figura 17. Variación del DMOS en función de la tasa de bits de codificación

Sin embargo, a la hora de degradar las secuencias de video, el sistema descrito más adelante en 3.3, solo funciona con secuencias hasta codificadas a 3 Mbps, ya que para tasas mayores, la herramienta para simular perdida de paquetes (Shunra, 3.3.4) no puede procesar los paquetes y se pierden fragmentos de las secuencias<sup>2</sup>.

Es también por este motivo que se optó por codificar los videos con un ratio constante de bits (CBR por su sigla en inglés), ya que al codificar los videos a un ratio variable de bits (VBR por su sigla en inglés) cuando el video tiene una cantidad de movimiento alta, se produce una ráfaga de bits que nuevamente hace saturar a la herramienta de pérdida de paquetes.

Es por todo lo anterior que efectivamente se codificó en CBR a 900 Kbits/s, 1.2 Mbits/s, 1.5 Mbits/s, 2.0 Mbits/s y 3.0 Mbits/s.

Otro parámetro que se definió es el Grupo de Imágenes (GOP por su sigla en inglés - ver 2.2). Para este parámetro se optó por el valor por defecto, es decir compuesto por quince cuadros, diez de los cuales son cuadros B. La composición del GOP es:

I B B P B B P B B P B B P B B

En lo que tiene que ver con el entrelazado, se optó por el barrido progresivo, de lo contrario, al reproducir los videos en formato digital se produce un efecto que degrada la imagen (especialmente en los bordes) y afecta negativamente al DMOS. Ver Figura 18.

<sup>2</sup> Antes de descartar la codificación a 6 Mbp/s, se recorrió un largo camino (descrito en 3.3.4) a lo largo del cual se probó cada herramienta a tasas mayores de 3 Mbp/s, no pudiendo lograr el streaming de las secuencias a través de estas herramientas.

El entrelazado busca mantener una buena calidad optimizando el ancho de banda disponible para transmitir las señales de TV. Las líneas entrelazadas de cada imagen están numeradas consecutivamente y divididas en dos campos: el campo "impares", que consiste en las líneas de los números impares y el campo "pares", que consiste en las líneas de los números pares. NTSC y PAL son formatos entrelazados.

En los sistemas de barrido progresivo, en cada período de refresco se actualizan todas las líneas [32].



Figura 18. Efecto del entrelazado

Por último, en cada caso se seleccionó la cantidad de cuadros por segundo correspondiente a cada formato de imagen. Para las secuencias en 525, 30 cuadros por segundo, mientras que para las secuencias en 625, 25 cuadros por segundo.

### 3.2.3. Porcentaje y modelo de pérdida de paquetes

Definida la forma en que se codifican las secuencias, es necesario definir la forma en que las secuencias se degradaran por su transmisión sobre una red IP y la consiguiente pérdida de paquetes.

Las pérdidas de paquetes que se escogieron son 0.5%, 1.0%, 2.0% y 5.0%.

Estos porcentajes se determinaron luego de varios ensayos, donde 5.0% resultó ser el máximo porcentaje de pérdida de paquetes para el cual la degradación hacía mínimamente aceptable la calidad del video degradado.

Como se puede observar se concentran más porcentajes de pérdida de paquetes en los valores más bajos, donde la degradación es menor. Esto se debe

a que en los ensayos preliminares se detectó que el DMOS es más sensible a las variaciones en bajas pérdidas (cuando las pérdidas son altas, el DMOS no varía mucho, se mantiene relativamente malo).

Por otra parte, el modelo de pérdida de paquetes seleccionado es el modelo de Bernoulli, es decir, una pérdida de paquetes aleatoria según el porcentaje de probabilidad de pérdida [33]. Si bien los modelos de Gilbert y Gilbert Eliot modelan de mejor manera la pérdida de paquetes en las redes IP. A los efectos de este estudio, configurar una pérdida de paquetes aleatoria según una cierta probabilidad de pérdida de paquetes resulta suficiente.

Los modelos de Gilbert y Gilbert Eliot sirven para modelar las pérdidas de paquetes por ráfagas [33]. Realizando ensayos para valores de largo medio de ráfaga entre 2 y 5 paquetes, siempre manteniendo una probabilidad de pérdida total constante, se comprobó que la degradación de la imagen por pérdida de paquetes, desde el punto de vista de la calidad percibida, es similar que la obtenida si perdemos paquetes en forma aleatoria según una probabilidad de pérdida (modelo de Bernoulli) igual a la probabilidad de pérdida total de los modelos antes referenciados.

Al obtenerse entonces resultados similares con todos los modelos, se seleccionó el modelo de Bernoulli por ser el más sencillo, y para disminuir las variables del estudio.

#### 3.2.4. Selección de secuencias degradadas

Es claro que degradando todas las secuencias de referencia, variando todos los parámetros en todos sus valores, produciría mucha información redundante desde el punto de vista estadístico. Es por esto que se procedió a confeccionar una matriz de secuencias degradadas que cubra todos los casos de interés.

En la *Tabla 2* se observa la selección de secuencias degradadas. Las entradas de la tabla marcadas con una “x” indican que se generó esa secuencia para ser utilizada tanto en las pruebas subjetivas como en las objetivas<sup>3</sup>.

---

<sup>3</sup> En la matriz se observan entradas marcadas en rojo. Estas secuencias no se pudieron generar por limitaciones de la herramienta de codificación ProCoder3. Esta herramienta no puede codificar secuencias de muy alta cantidad de movimiento a una tasa muy baja y constante (CBR).

		Codec - MPEG2																								
		Bit rate																								
		900 kbps					1.2 Mbps					1.5 Mbps					2.0 Mbps					3.0 Mbps				
		Prob. Perdida					Prob. Perdida					Prob. Perdida					Prob. Perdida					Prob. Perdida				
Source	Cantidad de movimiento	0.5	1	2	5	0.5	1	2	5	0.5	1	2	5	0.5	1	2	5	0.5	1	2	5					
1 Tree	Baja																									
2 Barcelona	Media		x					x						x	x						x					
3 Harp	Media		x					x						x	x						x					
4 Moving graphic	Baja			x					x	x							x				x					
5 Canoa Valsesia	Alta	x					x						x					x	x							
6 F1 Car	Alta			x						x								x			x					
7 Fries	Media			x						x	x							x			x					
8 Horizontal scrolling 2	Alta				x	x							x						x							
9 Rugby	Alta				x	x												x			x					
10 Mobile&calendar	Media	x					x						x							x	x					
13 Baloon-pops	Alta	x					x						x							x	x					
14 NewYork 2	Baja		x					x						x	x						x					
15 Mobile&Calendar	Media				x	x											x				x					
16 Betes_pas_betes	Baja	x					x						x							x	x					
17 Le_point	Alta			x					x	x							x				x					
18 Autumn_leaves	Baja		x						x					x	x						x					
19 Football	Alta		x						x					x	x						x					
20 Sailboat	Baja				x	x							x					x			x					
21 Susie	Baja			x					x	x								x			x					
22 Tempete	Media	x	x						x					x	x						x					

Tabla 2. Selección de secuencias degradadas

El primer paso para confeccionar esta tabla consistió en la clasificación de las secuencias de referencia. Las mismas fueron clasificadas según su formato y su cantidad de movimiento.

Las secuencias en formato PAL (625 líneas) van desde la secuencia 1 a la 10 de la matriz, mientras que las secuencias en formato NTSC (525 líneas) van desde la secuencia 13 a la 22.

Como se mencionó, también fueron clasificadas según su cantidad de movimiento. Para esto es les asignó una de tres categorías que son “Baja”, “Media” y “Alta”.

En base a esta clasificación, y a los parámetros que se deben variar, se marcaron con x las entradas de la tabla de forma de cubrir todas las combinaciones de formato, cantidad de movimiento, tasa de codificación y probabilidad de pérdida de paquetes.

### 3.3. Generación de videos degradados

Procesar los videos de referencia, de modo de obtener los pares de videos original/degradados, forma una parte sustancial de este trabajo. Degradar de una forma controlada y precisa las secuencias de video resulta vital, ya que es de estos pares de secuencias que se “alimentan” tanto las pruebas subjetivas como las objetivas. Si la degradación no es llevada a cabo de forma correcta, las conclusiones sobre la relación entre el porcentaje de pérdida de paquetes y la calidad percibida que se puede extraer de las pruebas no tienen ninguna validez.

El proceso al cual se somete a los videos de referencias para llegar a los pares de video original/degradado que se utilizan como entrada, tiene varias etapas. En la *Figura 19* se muestran en orden secuencial las distintas etapas del proceso de degradación.

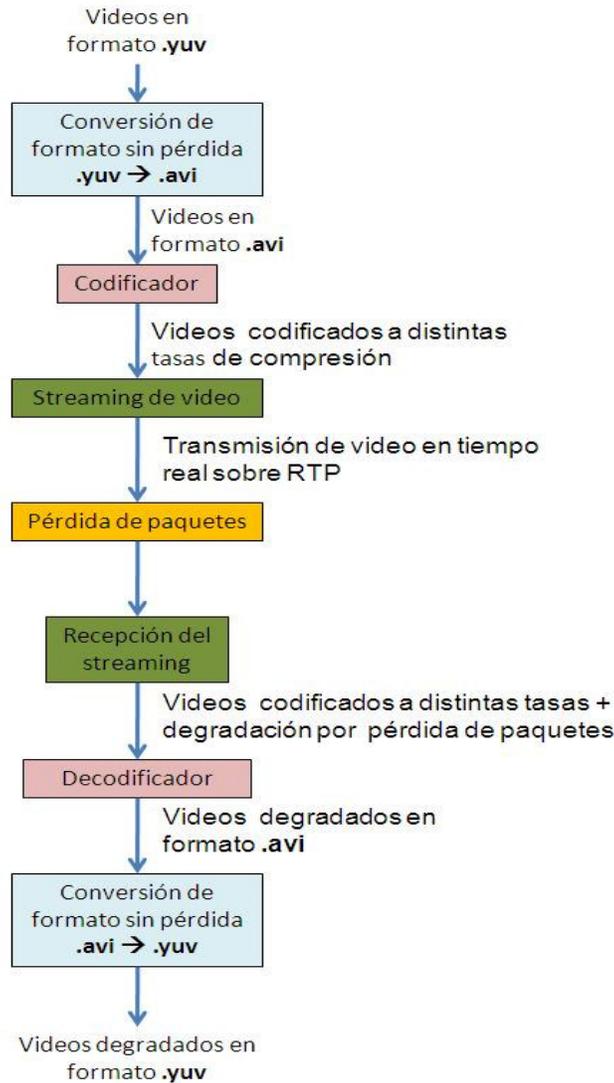


Figura 19. Etapas del proceso de degradación

A continuación se describen las herramientas con las que se implementa cada etapa, y la metodología que se emplea en cada una de ellas.

### 3.3.1. Conversión de videos en formato .yuv a formato .avi

Para esta tarea se utiliza el programa yuv2avi.exe © Streamcrest Associates. Este ejecutable corre sobre el sistema operativo Microsoft Windows y transforma los videos codificados en formato YUV descrito en [34], en archivos AVI, compatibles con las aplicaciones y codecs para Windows [35]. Este software es un freeware publicado por Streamcrest Associates y es el utilizado por varios miembros del VQEG<sup>4</sup>.

### 3.3.2. Codificación/Decodificación

La codificación de video se implementa con el software ProCoder3. Este software “combina la velocidad y la flexibilidad en una herramienta de conversión de vídeo racionalizado para profesionales” [36].

Es una herramienta de transcodificación ampliamente reconocida y premiada<sup>5</sup>, haciendo del software una herramienta muy confiable. Este punto no es menor, ya que si la herramienta de transcodificación no funciona correctamente, el resultado de este estudio se ve afectado haciéndolo inexacto.

El software corre sobre el sistema operativo Windows (sistema operativo sobre el cual corren todas las herramientas necesarias para este estudio) y tiene requerimientos mínimos ampliamente superados por el hardware con el que se cuenta.

Más allá de la de compatibilidad con el sistema operativo de elección y la posibilidad de ejecutarlo sobre el hardware con el que se cuenta, ProCoder3 ofrece, entre una gran lista algunas funcionalidades, algunas que resultan muy importantes para el estudio y para la ejecución del proyecto.

- Amplias características de entrada/salida. En particular una amplia librería de codecs, a cualquier tasa, tanto sea en CBR como en VBR.
- Procesamiento por lotes, codificar una sola fuente a múltiples objetivos al mismo tiempo, ejecutar múltiples trabajos de conversión de back-to-back. Estas funcionalidad permiten que el software procese varios videos en segundo plano, permitiendo avanzar en paralelo con otras tareas del proyecto

Para la decodificación de los videos utiliza también el ProCoder3.

---

4 En particular, este software fue recomendado a los autores de este proyecto por Fernando Ituo Higashie de CPqD - Diretoria de Inclusão e TV Digital, integrante del VQEG y participante del proyecto FRTV.

<sup>5</sup> Las premiaciones al ProCoder3 se pueden ver en <http://desktop.thomsongrassvalley.com/canopus/press/awards.php>

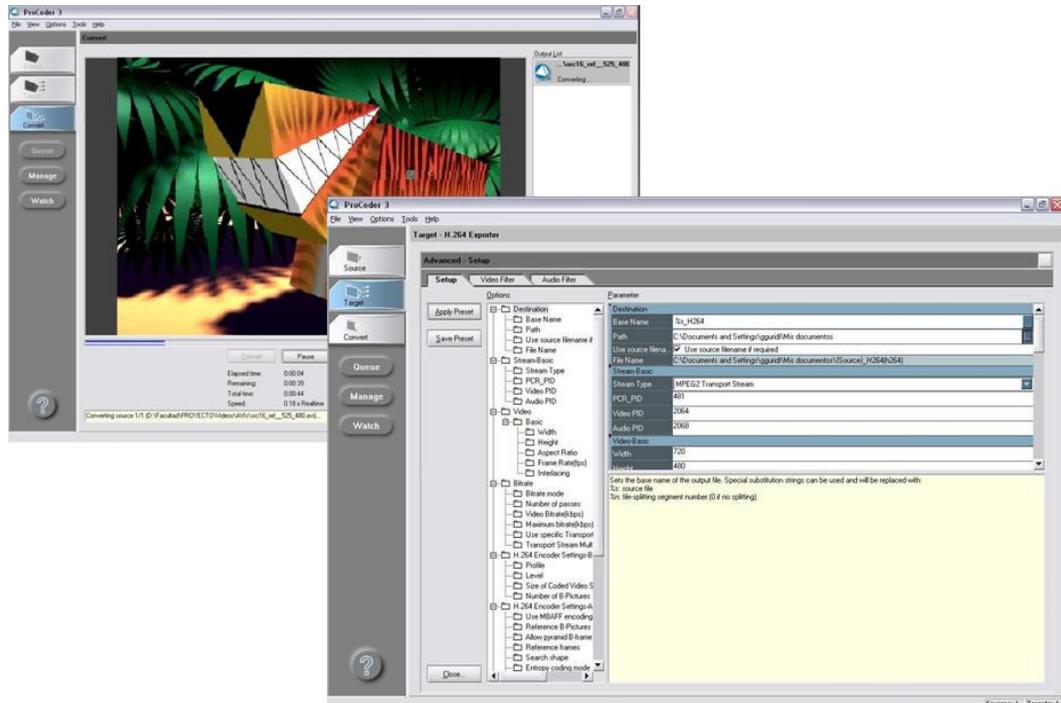


Figura 20. Vista de la herramienta ProCoder3

### 3.3.3. Streaming de video/recepción del streaming

Para simular los servicios de video en tiempo real sobre internet, se montó un ambiente para realizar streaming de video. Para esta tarea se utiliza la aplicación VLC. Este software surge en el marco del proyecto VideoLAN, este es un proyecto de software, que produce software libre y de código abierto para vídeo, publicado bajo la licencia GNU General Public License.

El software VLC media player es una aplicación freeware, consistente en un reproductor multimedia portátil para varios formatos de audio y vídeo (MPEG-1, MPEG-2, MPEG-4, DivX, mp3, ogg, etc) así como DVDs, VCDs, y varios protocolos de streaming [37].

Si bien no es necesaria la utilización de ningún codificador externo, en este proyecto se utilizó el ProCoder3 (3.3.2), para la codificación.

También puede ser usado como servidor para transmitir en unicast o multicast, en IPv4 o IPv6, en un alto rango de ancho de banda de red, y también como cliente de streaming.

Es precisamente esta funcionalidad la que permite, con muy baja complejidad, simular en laboratorio una transmisión de un streaming de video, siendo esta una de las razones por la cual esta herramienta fue seleccionada, ya que otros sistemas para hacer streaming de video requieren dos aplicaciones distintas, una como cliente y otra como servidor (por ej. Darwin Streaming server/QuickTime).

Por otra parte, el sistema de streaming de VLC (cliente /servidor) no altera de manera alguna el video transmitido al momento de encapsularlo en contenedor de

tipo MPEG-TS (MPEG Transport Stream), mientras que otros sistemas si lo hacen, por ejemplo, el Darwin Streaming Server modifica la cantidad de cuadros por segundo al encapsular el video.

Por las razones expuestas y por su alta flexibilidad es que el sistema de streaming seleccionado para este proyecto es VLC media player.

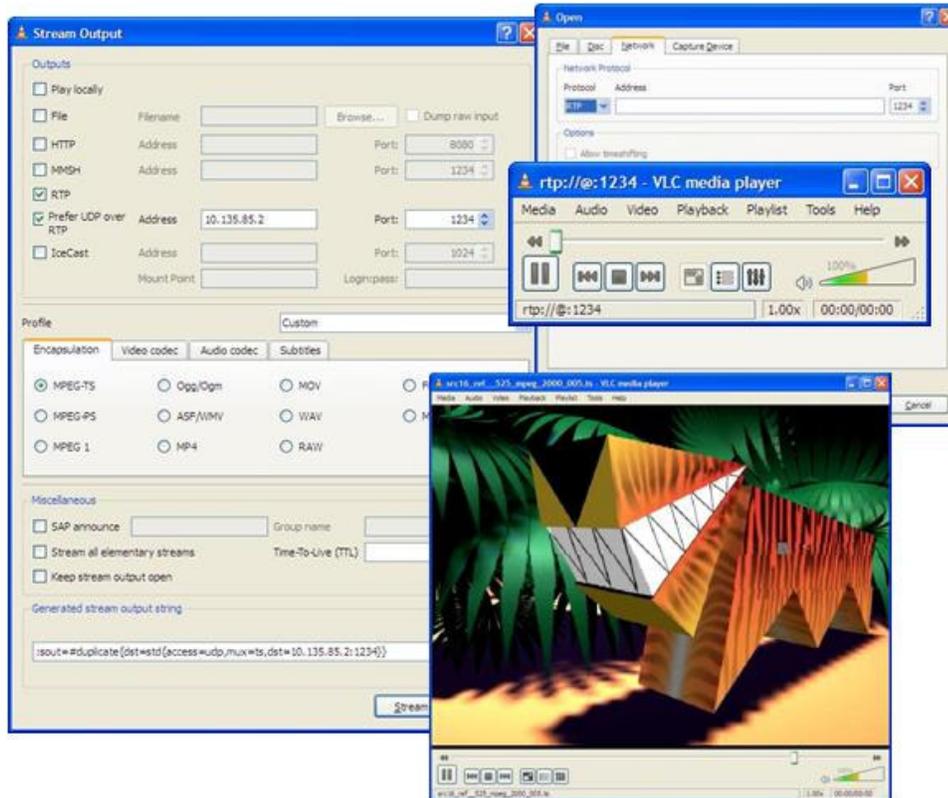


Figura 21. Vista de la herramienta VLC

Para la recepción del flujo de streaming se utiliza también el VLC media player.

### 3.3.4. Perdedor de paquetes

Como es sabido, este trabajo consiste en estudiar el comportamiento de los algoritmos de estimación de calidad percibida de video estandarizados en las normas [1] y [21] en un ambiente de transmisión en redes IP con pérdidas de paquetes. Por lo que el sistema utilizado para lograr ese ambiente de perdidas controlado era crucial.

Para esta tarea, la herramienta que se utiliza es el software SHUNRA VE SMB EDITION Versión 4.1 [38].

Este software es un software licenciado, aunque para la ejecución de este proyecto se utiliza un free trial (prueba gratis) que resulta suficiente. La versión que se utiliza corre sobre el sistema operativo Windows, y tiene una interfaz gráfica muy amigable.

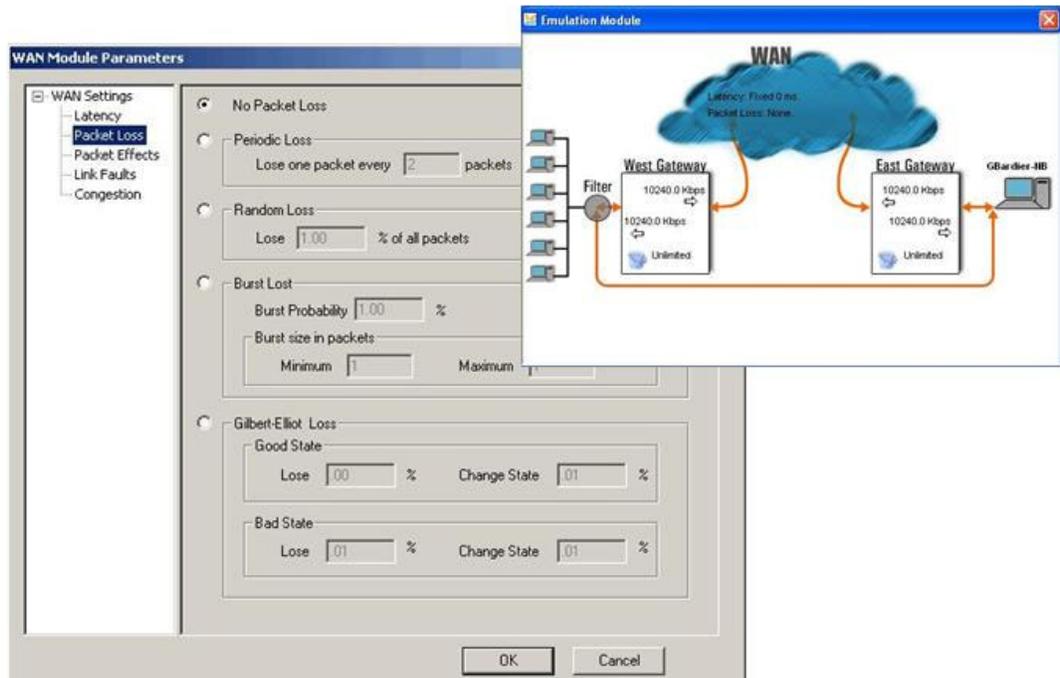


Figura 22. Vista de la herramienta Shunra

La elección de la herramienta no solo se basa en la compatibilidad con el sistema operativo de elección. Previamente, se recorrieron otras herramientas, como por ejemplo el algoritmo generado en JAVA por un proyecto anterior [39] y el Opnet IT Guru student version [40], pero por su simplicidad, robustez y bajo consumo de recursos (espacio libre en disco, memoria RAM y procesador), el Shunra fue seleccionado.

La utilización de esta herramienta simplifica también la topología del ambiente de laboratorio, pudiéndose ejecutar en forma corcoidente con el servidor de streaming seleccionado. Ver *Figura 23*.



Figura 23. Arquitectura del laboratorio para la degradación de los videos

A pesar que este software es un software reconocido<sup>6</sup>, el software se sometió a pruebas para validar su funcionamiento. Para ello se enviaron paquetes ICMP echo request (ping) por tiempos similares a los que insume transmitir las secuencias de referencia, obteniendo los porcentajes de pérdidas de paquetes esperados.

Si bien se trabaja en el modo avanzado [38], los ajustes para realizar pérdidas de paquetes controladas son bastante simples. La configuración consiste en declarar enlace Ethernet/IP entre dos máquinas y luego se define el tipo de condiciones a simular, por ejemplo latencia, pérdida de paquetes IP, jitter, congestión, etc.

Para este estudio, el módulo que reviste importancia es el de pérdida de paquetes. Para simular este comportamiento, el software tiene varios modelos, como por ejemplo los modelos de Gilbert y Gilbert Eliot. Para este estudio, configurar una pérdida de paquetes aleatoria según una cierta probabilidad de pérdida de paquetes resulta suficiente (3.2.3), siendo más sencillo de configurar.

### 3.3.5. Conversión de Formato sin Pérdida

La última tarea necesaria para generar los videos fue la conversión de los videos en formato AVI al formato YUV definido por el VQEG en [34]. Dado que no fue posible encontrar una herramienta que realizara esta función, se desarrolló una rutina de MATLAB que realiza la conversión, la cual consiste en tomar cada cuadro del video en formato RGB, cambiar este formato a YUV, sub-muestrear las crominancias, ya que el formato especificado por el VQEG es 4:2:2, y por último reordenar los valores. Los valores de los distintos cuadros son escritos uno a continuación del anterior sin insertar ningún separador ni referencia adicional, formando un archivo que contiene exclusivamente los valores de los pixeles de todos los cuadros.

El código de esta rutina se puede ver en el *Anexo A. Código fuente del algoritmo avi2yuv*.

---

<sup>6</sup> Ha sido utilizado por ejemplo por Stefan Winkler y Ruth Campos para su trabajo Video Quality Evaluation for Internet Streaming Applications [52].

### 3.4. Descripción de las pruebas subjetivas

Las pruebas subjetivas son la piedra angular de este estudio, dado que brindan el marco de referencia y representan la opinión de la media poblacional (valores “reales”).

Para realizar estas pruebas se seleccionó el método a utilizar siguiendo la tabla 2 descrita en el numeral 3 de la recomendación ITU-R\_BT.500-11 [13], “*Selección de métodos de Tests*”.

En dicha tabla se aconseja la utilización del método de Doble Estímulo Escala de Calidad Continua-DSCQS (Double Stimulus Continuous Quality Scale), para pruebas del tipo de referencia contra sistema degradado. Este método es explicado en el ítem 5 de [13], es el mismo método utilizado por el VQEG en la fase II. Esto es útil a la hora de comparar este estudio con el llevado adelante por el VQEG.

Otra de las características del método que resulta de interés para este estudio, es la baja dependencia a los efectos contextuales.

#### 3.4.1. Montaje del ambiente de prueba

El ambiente de pruebas se generó siguiendo las recomendaciones del párrafo 2.1.1 de la ITU-R-BT.500-11 [13] y la ITU-T P.910 [12]. Si bien el ambiente de pruebas no fue montado dentro de un laboratorio per se, se optó por este ambiente por ser el más estricto en las especificaciones. En las figuras *Figura 24*, *Figura 25* y *Figura 26* se puede observar el ambiente utilizado. En el monitor, se puede ver la imagen de uno de los sistemas PLUGE utilizado para calibrarlo.



Figura 24. Vista frontal 1



Figura 25. Vista frontal 2



Figura 26. Vista general del ambiente de pruebas

Se puede observar la mesa instalada en frente del monitor, la misma permito a los observadores utilizarla para marcar sus apreciaciones.

A continuación se observan las características del monitor utilizado.

Especificación.	Valor
Marca y modelo.	Viewsonic E90F+
CRT tamaño (medida en diagonal del área activa)	19"
Resolución (TVL)	>1000.
Dot-pitch.	0.25 mm

Tabla 3. Características del monitor



Figura 27. Vista frontal del monitor



Figura 28. Vista del luxómetro

También se midieron algunas de las características lumínicas del ambiente de pruebas, utilizando un luxómetro marca Sper / Scientific, modelo 840020. El equipo puede observarse en la *Figura 28*.

Las medidas recabadas se detallan a continuación.

Medidas.	Valor.
Luminancia de la pantalla inactiva en ambiente en condiciones de prueba.	0.1 cd/m <sup>2</sup>
Luminancia de la pantalla en blanco utilizando el sistema PLUGE en un ambiente oscuro.	71,4 cd/m <sup>2</sup>
Luminancia de la pantalla en negro en un ambiente oscuro.	0,1 cd/m <sup>2</sup>
Luminancia detrás de la pantalla con ambiente en condiciones de prueba.	2,7 cd/m <sup>2</sup>

Tabla 4. Valores de luminancia relevados

El brillo y contraste del monitor fue configurado mediante PLUGE (Picture Line-Up Generation Equipment) siguiendo las normas ITU-R BT.814 [41] y ITU-R BT.815 [42].

Hoy en día existen una variedad de sistemas PLUGE mediante software. Para esta tarea se utilizaron varios software, como Testscreens.exe [43] [7], "Monitor Test v3.0" [44] y "Nokia Monitor Test" [45]. Algunos valores no se pudieron medir por no contar con los instrumentos adecuados para ello, como por ejemplo la cromaticidad del Fósforo del monitor.

Respetando las recomendaciones de las normas, los observadores se ubicaron en un ángulo de observación menor a 30°, por lo que no se colocaron más de dos (2) observadores simultáneamente. Los mismos se ubicaron a una distancia de 5.H (=1,40 metros), siendo H la altura del monitor (28 cm), esto de acuerdo a las reglas PVD (Preferred Viewing Distance). El monitor acorde a las recomendaciones, no contaba con procesamiento interno de imágenes.

El contraste máximo del monitor no está expresado en el manual del equipo, pero se entiende que manteniendo un ambiente de observación tenue sin focos ni reflejos directos sobre el monitor, este llega a su nivel óptimo de visualización.

### 3.4.2. Armado de las secuencias de video

El armado de secuencias de pruebas consiste en concatenar pares de videos fuente y procesado, teniendo en cuenta que la cantidad total de videos observada por cada sujeto no debe superar dos sesiones de media hora.

Se muestran, consecutivamente, cada par de videos fuente y procesado, dos veces consecutivas. Entre video y video, se insertan dos (2) segundos donde se identifica el video a mostrar mediante una letra "A" o "B", y al final de cada par de videos mostrados dos veces, se introducen cuatro (4) segundos, para permitir al sujeto expresar en papel su calificación para cada video.

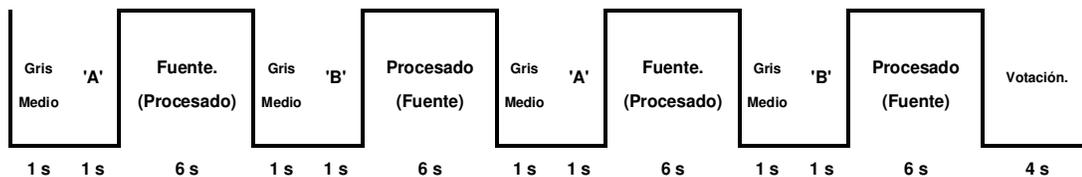


Figura 29. Composición de las secuencias de videos

El sujeto no tiene información previa de cual video es el fuente y cual el procesado. Esta definición persigue el no inducir la calificación del sujeto con los videos mostrados previamente.

Las sesiones de pruebas, no deben sobrepasar las dos (2) “medias sesiones” de 30 minutos cada una, ya que más tiempo se considera que agota la visión del observador y atenta sobre la concentración, pudiendo reflejarse en menores calificaciones para los últimos videos.

Se separó en dos sesiones correspondientes a los formatos estudiados, PAL N y NTSC. La primer sesión del formato 525@ 60Hz contaba con 30 secuencias de videos, tres al inicio con el fin de entrenar al observador, cuyos resultados se descartaron. La segunda sesión contaba con 26 secuencias en el formato 625@ 50Hz, los tres primeros resultados se descartaron por corresponder también a secuencias de entrenamiento. Se generaron dos secuencias completas de cada formato, teniendo especial consideración en no generar efectos contextuales generados por el orden de los videos.

Para la generación de estas cuatro secuencias de video (dos en PAL N y dos en NTSC) se utilizó la herramienta Vegas Movie Studio 9™ de Sony®.

Con el software Vegas Movie Studio™, las secuencias de video se editan simplemente haciendo “drag-and-drop” y ejecutando comandos básicos como cut, copy y paste, para crear secuencias profesionales. Esta herramienta funciona sobre el sistema operativo de elección (Windows) y corre en el hardware con el que se cuenta.

Vegas Movie Studio™ provee también el DVD Architec Studio™, para crear DVD’s con las secuencias generadas de forma sencilla. DVD Architect Studio™ soporta los formatos de video más populares como AVI, MPEG-1, MPEG-2, etc. [46]

Si bien este software es pago, los treinta días de la versión de prueba fueron suficientes para la confección de las cuatro secuencias antes mencionadas.

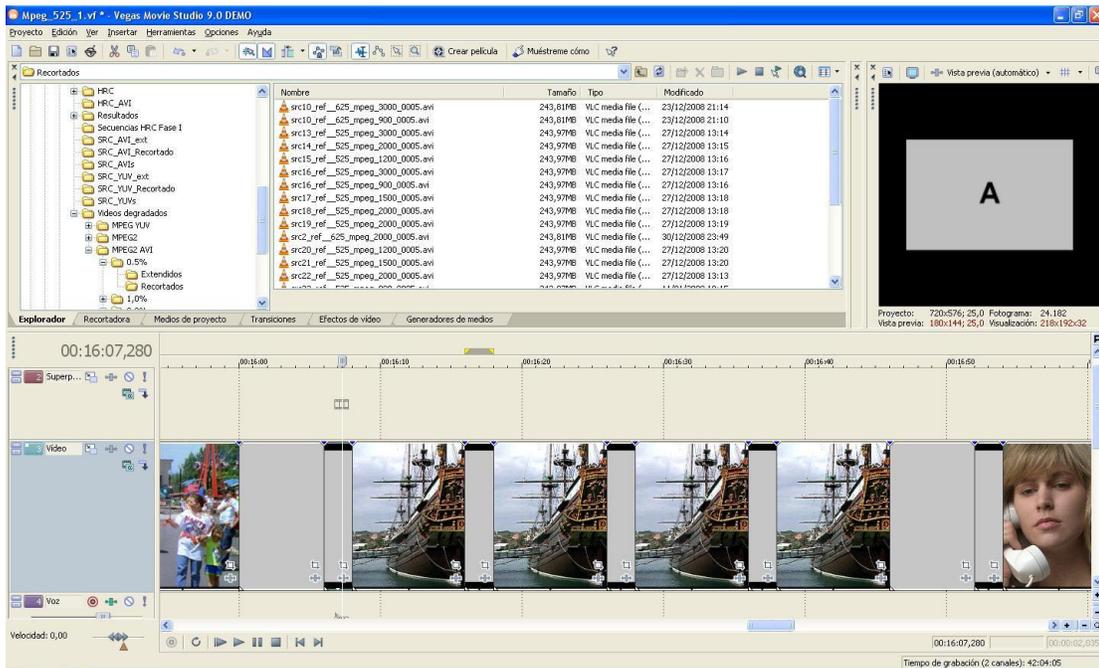


Figura 30. Vista de la herramienta Vegas Movie Studio

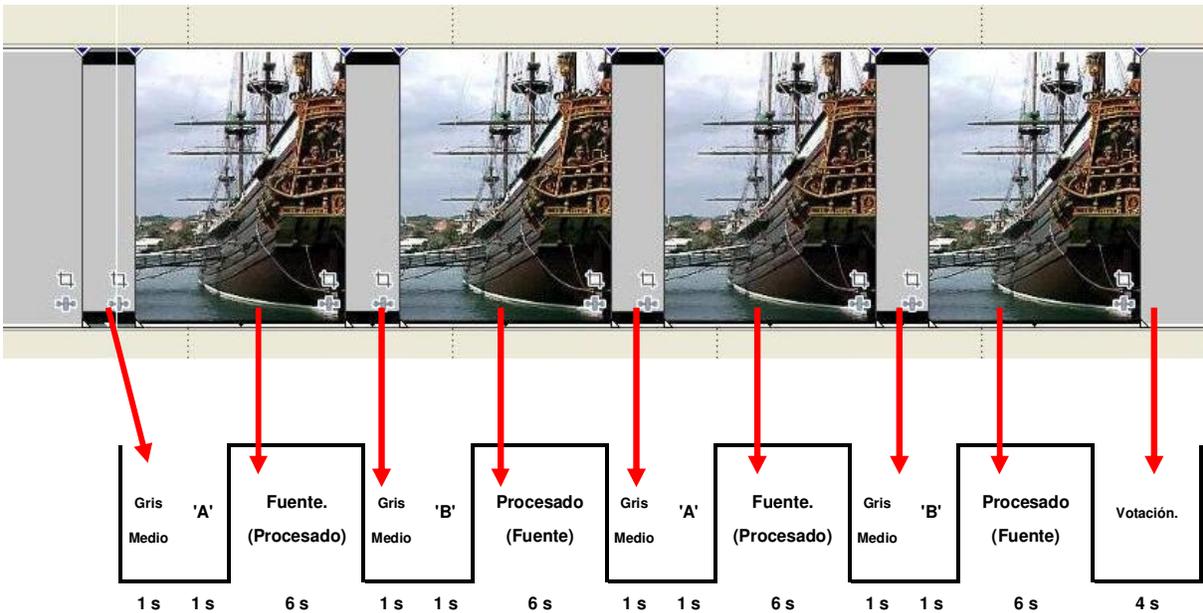


Figura 31. Visualización del armado de las secuencias con la herramienta Vegas Movie Studio

En el Anexo B. *Composición de las secuencias de video*, se puede observar la composición de cada una de las secuencias de videos editadas.

### 3.4.3. Dimensionamiento de las pruebas

La realización de las pruebas subjetivas busca obtener para cada video un valor de DMOS, valor que representa la opinión de la media poblacional acerca de la calidad del video. Al estimar este valor con la opinión de un número finito de individuos se comete un error, y el valor obtenido no es exactamente el de la media poblacional. Parece claro que cuanto más grande sea la muestra más cerca estará el valor estimado a través de las pruebas del valor real<sup>7</sup>. Inmediatamente surge la pregunta de, ¿cuál será la cantidad necesaria de sujetos para tener una buena aproximación?

Para este estudio, la estimación del número de individuos necesarios resulta de suma importancia, este número impacta directamente en la duración de la etapa de pruebas subjetivas.

La pregunta planteada es un problema clásico de la estadística que puede ser abordado mediante la estimación de intervalos de confianza.

El problema puede plantearse en forma más general como un problema de 4 parámetros, en el cual tres son datos y el cuarto es una incógnita, siendo estos parámetros el error máximo sobre la magnitud a estimar, la desviación estándar poblacional de la magnitud, el número de individuos en la muestra y el nivel de confianza del valor estimado [47].

<sup>7</sup> Si bien se asume la existencia de un valor de DMOS real, el mismo no puede ser calculado, ya que implicaría conocer la opinión de todo el universo del experimento, que para este estudio, coincide con toda la población del planeta. De todas formas, la teoría estadística permite trabajar sin conocer este valor.

La ecuación que relaciona estos parámetros es la siguiente:

$$n = \left( \frac{Z_{\alpha/2} \sigma}{E} \right)^2$$

Ecuación 1

Donde

$n$  es el número de individuos necesarios

$Z_{\alpha/2}$  es el valor de abscisas que deja una cola de área  $\alpha/2$  a su derecha en la distribución Normal (0,1). Si queremos un nivel de confianza  $I$ ,  $\alpha = 1 - I$

$\sigma$  es la desviación estándar

$E$  es el error máximo permitido en la estimación

El intervalo de confianza y el error máximo sobre la magnitud a estimar son parámetros que se pueden elegir, mientras que la desviación estándar de la media poblacional es un dato del problema. Al no contar con datos de la desviación estándar del DMOS para experimentos como este, es imposible calcular a priori el tamaño de la muestra necesario

Para resolver esta situación se utilizaron los datos del experimento del VQEG fase I, el cual en su informe final brinda los datos del DMOS y su desviación estándar, teniendo claro que este experimento no contempla la pérdida de paquetes, y por tanto, las características de los datos pueden ser distintas, en particular la desviación estándar. Lo anterior, sirvió para realizar un cálculo primario del tamaño de la muestra. Una vez realizadas las pruebas subjetivas, se recalculó el tamaño de la muestra necesario, pero esta vez con los valores de desviación estándar obtenidos en condiciones de pérdida de paquetes.

La desviación estándar promedio hallada para los valores surgidos de la fase I del proyecto FRTV del VQEG son de 0,1588 para los videos en formato NTSC, y de 0,1719 para los videos en formato PAL.

Los valores obtenidos para los distintos parámetros de entrada se pueden ver en las siguientes tablas, en las cuales se indica la cantidad de sujetos necesarios en función del error y del nivel del intervalo de confianza.

525		Error en el DMOS				
		0,005	0,01	0,02	0,05	0,1
Intervalo de confianza en sigmas	1	1008,166	252,041	63,010	<b>10,082</b>	2,520
	2	4032,662	1008,166	252,041	40,327	10,082
	3	9073,490	2268,373	567,093	90,735	22,684

Tabla 5. Sujetos necesarios para el formato NTSC según los datos obtenidos del FRTV fase I

625		Error en el DMOS				
		0,005	0,01	0,02	0,05	0,1
Intervalo de confianza en sigmas	1	1182,364	295,591	73,898	<b>11,824</b>	2,956
	2	4729,457	1182,364	295,591	47,295	11,824
	3	10641,279	2660,320	665,080	106,413	26,603

Tabla 6. Sujetos necesarios para el formato PAL según los datos obtenidos del FRTV fase I

En negrita se resaltan los valores que se estimaron adecuados, ya que se obtiene un error aceptable, con un número de individuos razonable.

Una vez realizadas las pruebas, se procedió a recalcular la cantidad de individuos necesarios, utilizando esta vez la desviación estándar surgida de las pruebas (0,1905 para videos en formato NTSC y 0,1863 para videos en formato PAL), para obtener los resultados con los niveles de confianza apropiados. Si bien fue necesario incrementar el número de sujetos, la estimación inicial fue buena.

Una vez recalculada la cantidad de sujetos necesarios, se procedió a realizar las pruebas a los sujetos adicionales

Los resultados obtenidos fueron los siguientes:

<b>525</b>		Error en el DMOS				
		0,005	0,01	0,02	0,05	0,1
Intervalo de confianza en sigmás	1	1452,032	363,008	90,752	<b>14,520</b>	3,630
	2	5808,127	1452,032	363,008	58,081	14,520
	3	13068,287	3267,072	816,768	130,683	32,671

Tabla 7. Sujetos necesarios para el formato NTSC para este estudio

<b>625</b>		Error en el DMOS				
		0,005	0,01	0,02	0,05	0,1
Intervalo de confianza en sigmás	1	1388,682	347,171	86,793	<b>13,887</b>	3,472
	2	5554,728	1388,682	347,171	55,547	13,887
	3	12498,138	3124,535	781,134	124,981	31,245

Tabla 8. Sujetos necesarios para el formato PAL para este estudio

Como se deduce de lo anterior, cada video en formato PAL N (625 líneas) debió ser observado por 15 personas para obtener un valor con menos de un 5% de error, dentro de un intervalo de confianza de 1 sigma. Por otra parte, cada video en formato NTSC (525 líneas) debió ser observado por 14 personas para obtener un valor con menos de un 5% de error, dentro de un intervalo de confianza de 1 sigma.

La diferencia entre la cantidad de sujetos necesarios para los videos en formato PAL y formato NTSC probablemente son provocadas por las diferencia en la muestra, ya que a excepción de una secuencia, las secuencias de referencia en formato PAL son distintas a las de formato NTSC.

### 3.4.4. Observadores

Está previsto que los observadores no poseerán conocimiento directo de temas relacionados a la calidad de video. En este caso, si bien muchos de los observadores son estudiantes universitarios, ninguno posee experiencia o conocimientos previos sobre calidad de video.

Se realizaron pruebas de visión, pruebas de distinción del color (Daltonismo) y pruebas de contraste. Las pruebas de visión se realizaron mediante los test comunes, empleados por los oftalmólogos, como el Ishihara Test de distinción del color [48].

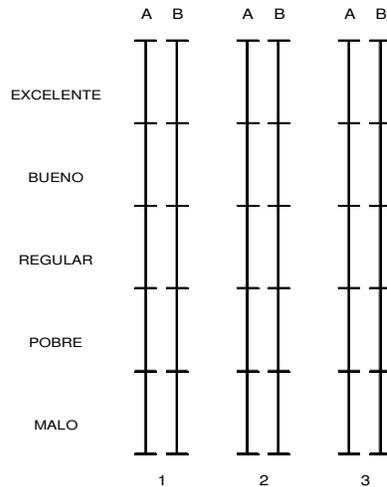


Figura 32. Escala para la calificación de videos

Los observadores deben calificar cada video observado mediante una marca en una escala continua como la mostrada en la *Figura 32*, aunque de mayores dimensiones.

La escala original mide 10 cm y las marcas son exactamente cada 2 cm. Las escalas se imprimieron de a pares para facilitar el seguimiento de la sesión. Las marcas definen cinco regiones de calificación bien definidas a efectos de orientar la apreciación, Excelente, Bueno, Regular, Pobre y Malo.

A los observadores se les dieron instrucciones claras y concisas al comienzo de cada sesión. De todas maneras se entrenaron en la tarea de calificar videos visualizando algunos videos de muestra, a fin de comprender completamente lo que se les pedía realizar. También se previó que los sujetos necesitan cierta aclimatación al tipo de prueba, por lo tanto se dispusieron de tres videos iniciales solo a efectos de lograr esta aclimatación del sujeto. Las calificaciones de estos videos se descartaron.

El siguiente texto es entregado a cada observador al inicio de cada prueba. Para no repetir figuras no se incluyen todas las imágenes del texto original:

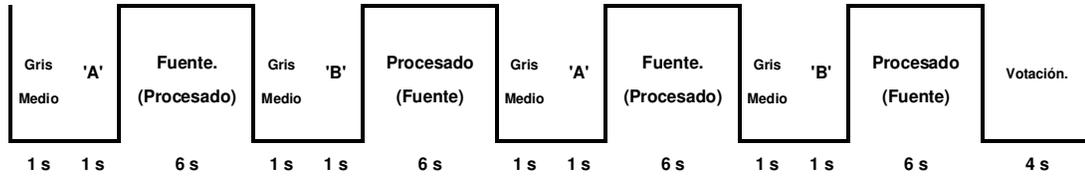
*"Buenas tardes.*

*En el test que haremos a continuación les pedimos que evalúen la calidad global de los videos que van a observar. Estamos interesados en su opinión de la calidad de cada escena. Por favor no juzguen el contenido de la escena o la calidad de la actuación. Tomen en cuenta diferentes aspectos de la calidad de video y formen su opinión basados en la impresión general de la calidad del video.*

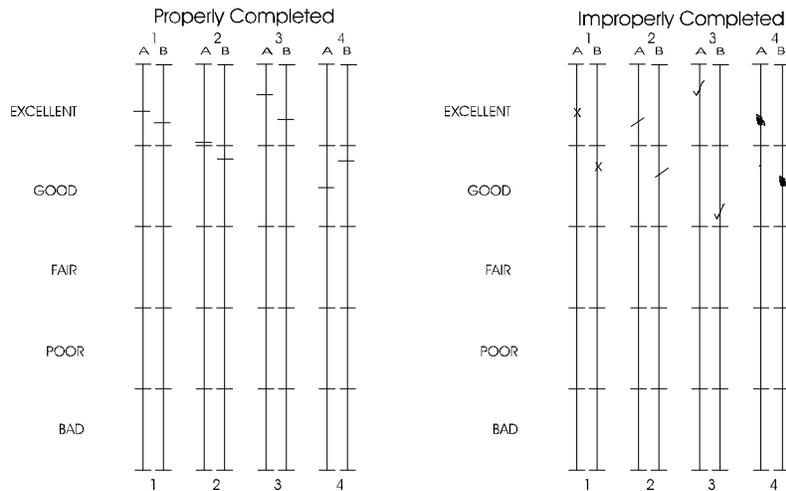
*Algunos de los posibles problemas en la calidad son:*

- *Reproducción pobre o inconsistente de los detalles.*
- *Reproducción pobre de los colores, brillo o profundidad.*
- *Reproducción pobre del movimiento*
- *Imperfecciones, como falsos objetos o efecto "nieve"*

*El test consiste en una serie de pruebas. Durante cada prueba, dos versiones de un mismo video, serán mostradas en la siguiente manera:*



"A" es la primera versión, "B" es la segunda versión. La primera presentación de cada prueba será anunciada con una letra A y la segunda con una letra B. Luego este par de videos será repetido completando una prueba. Cuando juzguen la calidad de las presentaciones les pedimos que utilicen escalas de juicio como la que se presenta a continuación:



Como pueden ver, hay dos escalas para cada prueba, una para la versión A y otra para la versión B, dado que ambas versiones deben ser juzgadas.

Las escalas de juicio son líneas verticales continuas que están divididas en cinco segmentos. Como una guía, los adjetivos, "excelente", "bueno", "regular", "pobre" y "malo" se han adjuntado a cada segmento. Les pedimos que ubiquen una sola línea horizontal en el punto de la escala que más se parezca a su juicio de la calidad general de la presentación, como se ve en la figura. [12]

Cada hoja de resultados se identificó con el formato y el número de secuencia proyectada (ver 3.4.2). Además, cada observador debía escribir su nombre completo y su edad.

En la práctica se tuvo que enfrentar diferentes errores e imprevistos. Algunos de estos errores son por ejemplo, que las personas, a pesar de las explicaciones, marcaron con cruces o varias rayas, esto se corrigió tomando el valor medio de las marcas. Otro error ocurrido varias veces, fue el corrimiento de las calificaciones, por ejemplo, marcaban en la escala del video 6 y realmente estaban calificando el video 5, esto ocurría porque las escalas eran quizás demasiado angostas. Este inconveniente no revisó importancia, ya que al llegar al final de la hoja de calificaciones, los observadores se daban cuenta por si mismos que les faltaba una escala, al volver para atrás era evidente en cual video ocurrió el corrimiento. Otro tipo de errores fue el de no considerar que los celulares debían estar apagados o en silencio durante toda la sesión, de otra manera esto generaba grandes distorsiones y en general debíamos recomenzar la sesión.

### 3.5. Descripción de los algoritmos para pruebas objetivas

Aquí se presentan los algoritmos desarrollados por el Centro de Pesquisa e Desenvolvimento em Telecomunicações (CPqD) y la National Telecommunications and Information Administration (NTIA) en el marco de la segunda fase del proyecto FRTV del VQEG. Estos algoritmos se presentan en forma resumida, sin adentrarse en profundidad ya que no es el objetivo de este estudio.

A su vez se introduce la definición de PSNR y se describe en profundidad la implementación del algoritmo de Yonsei.

El diseño y funcionamiento de cada uno de los algoritmos esta especificado en las normas ITU-T J.144 [21] y ITU-R BT.1683 [1].

#### 3.5.1. CPqD-IES

El modelo presentado por CPqD en la fase II del proyecto FRTV se denomina CPqD-IES versión 2.3. (IES por la sigla en inglés de "Image Evaluation based on Segmentation").

CPqD-IES v.2.3 implementa evaluación de la calidad del video usando parámetros objetivos, y basándose en la segmentación de las imágenes. Las escenas son segmentadas en planos, bordes, regiones de textura asignándosele a cada uno de estos contextos una serie de parámetros objetivos.

Para este modelo, la relación entre cada parámetro objetivo y la medición subjetiva de la degradación, es aproximada por una curva sigmoideal, resultando en un nivel estimado de la degradación para cada parámetro.

El resultado final es alcanzado a través de la combinación de niveles estimados de degradación, basados en su consistencia estadística.

Un clasificador de escenas es utilizado para ubicar las mismas y evaluarlas en forma independiente. Este clasificador utiliza información espacial (a través de la transformada DCT) e información temporal (basándose en los cambios de los segmentos) de la secuencia de entrada, para obtener un parámetros modelo desde una base de datos de escenas de referencia [16].

Dadas las características de este estudio, y que el mismo tiene fines exclusivamente académicos, este software (CPqD-IES versión 2.3) fue compartido por el Centro de Pesquisa e Desenvolvimento em Telecomunicações (CPqD).

En particular se compartió un archivo ejecutable en ambiente DOS, y un manual de usuario. En ningún momento se accedió al código del programa.

#### 3.5.2. NTIA

El software presentado por la National Telecommunications and Information Administration (NTIA) es el VQM.

El módulo para propósito general, desarrollado para trabajar en un amplio rango de calidades y tasas de bits, fue el evaluado en la fase II del FRTV del VQEG y a la postre estandarizado, si bien el software cuenta con módulos específicos para sistemas de televisión, video conferencia, y desarrollo (optimiza la velocidad de computo).

Este algoritmo está implementado de distintas formas para correr sobre sistemas operativos Windows o Linux, también para trabajar tanto desde línea de comandos o desde una interfaz gráfica de usuario (GUI por su sigla en inglés).

Para obtener las distintas versiones del software se puede obtener más información en [www.its.blrdoc.gov/n3/video/vqmssoftware](http://www.its.blrdoc.gov/n3/video/vqmssoftware)

En este estudio se utilizó la versión de propósito general para correr desde DOS. Este modelo ha sido diseñado y probado para trabajar con distintos codecs en un amplio rango de tasas de codificación (por ejemplo, tasas de bits desde 10 Kbits/s hasta 45 Mbits/s en MPEG-1/2/4) [16].

El procesamiento incluye el muestreo de la secuencia de referencia y la degradada, calibración de ambas secuencias, extracción de las características de cada secuencia, cálculo de los parámetros de la calidad del video y cálculo del modelo general.

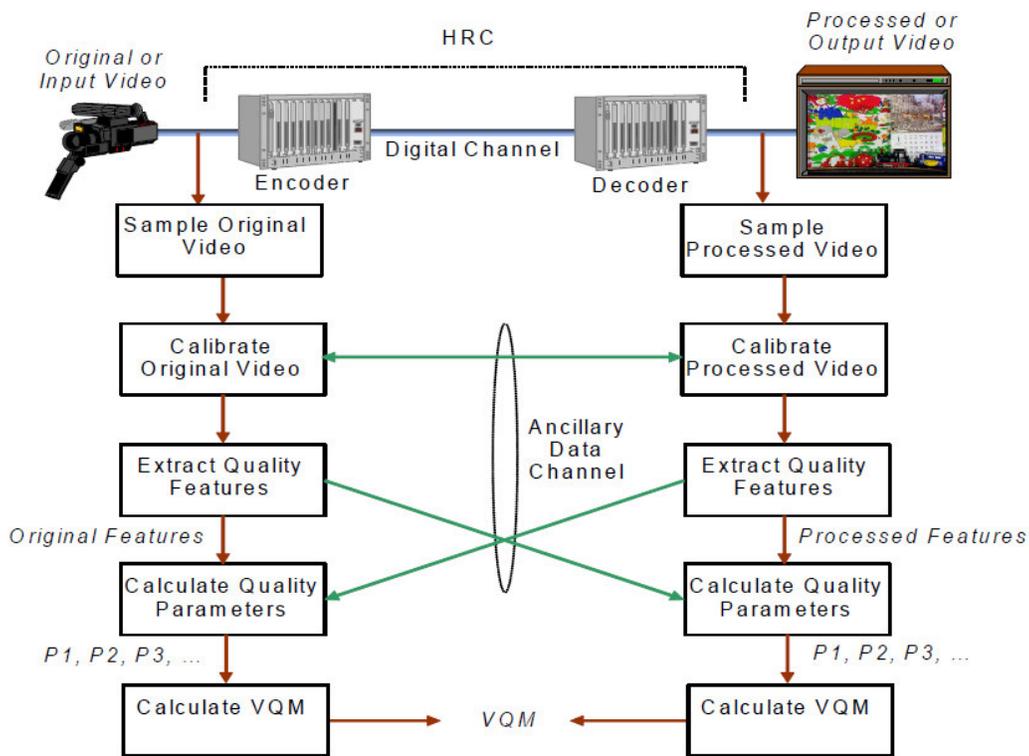


Figura 33. Pasos requeridos para el cálculo del VQM

En la *Figura 33* se puede observar un diagrama funcional del algoritmo.

El modelo general es sensible a cambios en la percepción de la calidad debida a las distorsiones provocadas por la codificación, el canal digital y la decodificación [21].

### 3.5.3. PSNR

La relación Señal a Ruido de Pico (PSNR por su sigla en inglés) se define como la relación entre la máxima energía posible de una señal y el ruido que la afecta. El PSNR suele utilizarse como medida cuantitativa de la calidad de video. Para este fin, su cálculo se realiza de la siguiente manera.

En primer lugar se haya el error cuadrático medio de cada cuadro.

$$EC^l = \sum_{i=1}^M \sum_{j=1}^N \{F^l(i, j) - P^l(i, j)\}^2$$

Ecuación 2

Donde:

$EC^l$ : Error Cuadrático de la imagen L.

$F^l(i, j)$ : Cuadro L del video fuente.

$P^l(i, j)$ : Cuadro L del video procesado.

M: Numero de filas.

N: Numero de columnas.

En segundo término, se calcula un error cuadrático medio entre la secuencia de referencia y la secuencia procesada.

El error cuadrático medio del video es:

$$ECM = \frac{1}{K} \sum_{l=1}^L EC^l$$

Ecuación 3

Donde:

$ECM$ : Error Cuadrático medio de todo el video.

$EC^l$ : Error Cuadrático de la imagen L.

L: Cantidad de cuadros en la secuencia analizada.

K: Cantidad total de píxeles de borde en toda la secuencia fuente.

Finalmente calculamos el PSNR de la siguiente manera:

$$PSNR = 10 \cdot \log_{10} \left( \frac{P^2}{ECM} \right)$$

Ecuación 4

Donde:

PSNR: Peak Signal to Noise Ratio (Relación señal a ruido).

P: Valor de pico de píxel (en este caso P=255).

ECM: Error Cuadrático medio.

Es importante mencionar que en este estudio se utilizan los valores de PSNR generados por el algoritmo por el VQM para ser comparado con los otros modelos y con el propio VQM.

### 3.5.4. Desarrollo del algoritmo de Yonsei

Al momento de decidir el lenguaje que se utilizaría para implementar el algoritmo se estudiaron las ventajas y desventajas de C++, Java y MATLAB. C++ y Java parecen más adecuados si se precisa un algoritmo rápido que pueda correr junto con alguna aplicación de visualización de multimedia en una PC de usuario. Java presenta la ventaja de ser multiplataforma, corriendo en los sistemas de diferentes usuarios. Finalmente, dado que este estudio se concentra en un análisis estadístico con videos fuentes y pre-procesados, no resultó de interés generar un algoritmo rápido, sino robusto y eficiente. Se decidió entonces por utilizar MATLAB y su simple lenguaje para la estructura de archivos “.m” de comandos secuenciales. Asimismo, MATLAB cuenta con una base importante de funciones de procesamiento de imágenes y videos.

Habiendo obtenido los algoritmos de la NTIA y de CPqD se decidió generar el algoritmo propuesto en [16] por la Universidad de Yonsei, Seoul [49], estandarizado en la recomendaciones ITU-T J.144 (03/2004) [21] y en [1]. Siguiendo lo indicado en estas normas, se buscó generar un código capaz de correr en un PC de escritorio y al mismo tiempo fuese lo suficientemente rápido para que permitiera trabajar en tiempos razonables para realizar tareas como la depuración del algoritmo<sup>8</sup>.

El modelo propuesto por la universidad de Yonsei se basa en la hipótesis de que la percepción del ojo humano es más sensible a degradaciones en los bordes de las imágenes. También se basa en que la mayoría de los codificadores de video hacen énfasis en la compresión en los bordes de las imágenes. Por lo tanto, el algoritmo realiza en primera instancia una detección de bordes en el video fuente, para luego calcular el error cuadrático medio entre ambas imágenes (original y degradada) en los píxeles de borde encontrados.

#### 3.5.4.1. Algoritmo

El algoritmo implementado funciona de la siguiente manera. En primera instancia se cargan en memoria de MATLAB ambos videos, fuente y procesado a comparar. Se toma un cuadro del video fuente originalmente en RGB255 y se convierte en formato YCbCr. El formato RGB255 se compone de una matriz de  $M \times N \times 3$ . Donde la primer matriz  $M \times N$ , representa el componente Rojo (R = Red), la segunda matriz representa el componente verde (G = Green) y la tercer matriz representa el componente azul (B = Blue). En el formato de NTSC 525@60Hz  $M=486$ ,  $N=720$  y en el formato de PAL 625@50Hz  $M=576$ ,  $N=720$ . Cada imagen en el formato YCbCr se compone de 3 matrices, la primera componente contiene la luminancia (Y), la segunda la crominancia azul (Cb) y la tercera la crominancia roja (Cr).

Luego el algoritmo procesa únicamente la componente de la luminancia (Y), aplicando el filtro de Sobel vertical y luego horizontalmente. Obtenida esta imagen

---

<sup>8</sup> Con videos en formato AVI de 160 Mb, los tiempos de procesamiento resultaron ser del orden de 160 segundos para cada par secuencia fuente y procesada.

de gradientes, mediante un umbral se consigue una matriz binaria que representan los bordes de la imagen. Esta máscara se utilizará para obtener un PSNR de borde.

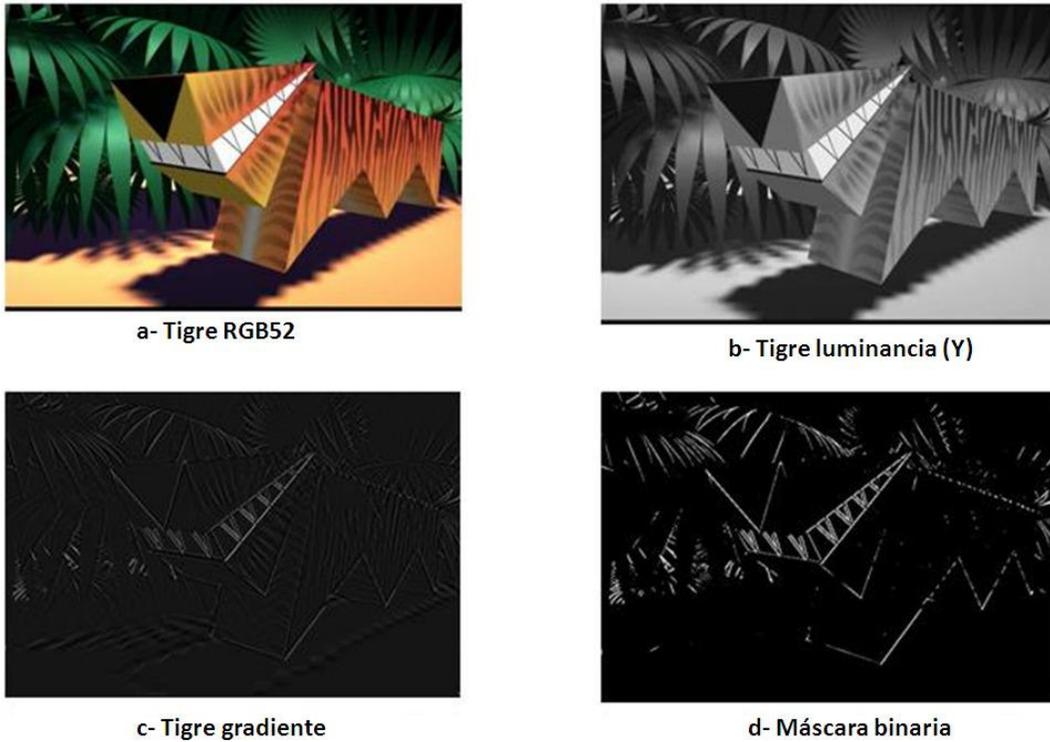


Figura 34. Visualización da las distintas etapas del tratamiento para cada cuadro en el algoritmo de Yonsei

$$EC_{borde}^l = \sum_{i=1}^M \sum_{j=1}^N \{F^l(i, j) - P^l(i, j)\}^2 \text{ si } |R^l(i, j)| \neq 0$$

Ecuación 5

Donde:

$EC_{borde}^l$ : Error Cuadrático de borde de la imagen L.

$F^l(i, j)$ : Cuadro L del video fuente.

$P^l(i, j)$ : Cuadro L del video procesado.

$R^l(i, j)$ : Cuadro L del video mascara generado.

$M$ : Numero de filas.

$N$ : Numero de columnas.

Utilizando la *Ecuación 5* se calcula un error cuadrático únicamente en los píxeles reconocidos como bordes.

El error cuadrático medio de borde se calcula de la siguiente manera:

$$ECM_{borde} = \frac{1}{K} \sum_{l=1}^L EC_{borde}^l$$

Ecuación 6

Donde:

$ECM_{borde}$ : Error Cuadrático medio de borde.

$EC_{borde}^l$ : Error Cuadrático de borde de la imagen L..

L: Cantidad de cuadros en la secuencia analizada.

K: Cantidad total de píxeles de borde en toda la secuencia fuente.

Finalmente el PSNR de bordes se calcula como:

$$EPSNR = 10 \cdot \log_{10} \left( \frac{P^2}{ECM_{borde}} \right)$$

Ecuación 7

Donde:

$EPSNR$ : Edge Peak Signal to Noise Ratio (Relación señal a ruido de borde).

P: Valor de pico de píxel (en este caso P=255).

$ECM_{borde}$ : Error Cuadrático medio de borde.

El Umbral utilizado fue de 260. Luego de realizarse varias pruebas con diferentes umbrales, se definió la utilización del mismo umbral utilizado previamente por la Universidad de Yonsei [49] en la fase II [16].

### 3.5.4.2. Post Ajustes

Una vez ejecutado el algoritmo es necesario procesar los resultados obtenidos de forma de ajustarlos para modelar de mejor manera la medición de la calidad percibida.

A continuación se describe este procesamiento en forma detallada.

#### **De-énfasis del EPSNR**

Cuando el valor del EPSNR es mayor a 35 se realizan ciertos ajustes descriptos a continuación:

$$EPSNR = \begin{cases} EPSNR & \text{si } 0 \leq EPSNR \leq 35. \\ EPSNR * 0,9 & \text{si } 35 \leq EPSNR \leq 40. \\ EPSNR * 0,8 & \text{si } EPSNR > 40. \end{cases}$$

Ecuación 8

### Bordes borrosos

Sabido es que si las imágenes presentan bordes borrosos la calidad percibida por los observadores tiende a disminuir. Este comportamiento se considera y se modela de la siguiente manera.

$$MEPSNR = \begin{cases} EPSNR - 60 * \left( 0.1225 - \left( \frac{PB_{comunes}}{PB_{fuente}} \right)^2 \right) & \text{si } EPSNR \leq 25 \text{ y } \frac{PB_{comunes}}{PB_{fuente}} < 0,35 \text{ y } \frac{PB_{procesado}}{PB_{fuente}} < 0,13. \\ EPSNR & \text{para cualquier otro caso.} \end{cases}$$

#### Ecuación 9

Donde:

$MEPSNR$ : Modified Edge Peak Signal to noise ratio. (PSNR de borde modificado).

$EPSNR$ : Edge Peak Signal to Noise Ratio (Relación señal a ruido de borde).

$PB_{comunes}$ : Píxeles de borde comunes a la fuente y al video procesado.

$PB_{fuente}$ : Píxeles de borde del video fuente.

$PB_{procesado}$ : Píxeles de borde del video procesado.

### Escalado

Finalmente, los resultados obtenidos luego de las etapas anteriores, se someten a un proceso de escalado de 0 a 1 donde 0 representa que son indistinguibles el video procesado con el fuente y 1 que son completamente diferentes, fuente y procesado.

El escalado se realiza de la siguiente manera:

$$VQM = 1 - 0,02 * MEPSNR$$

#### Ecuación 10

VQM es el resultado final del modelo generado.

### 3.5.4.3. Diagrama del algoritmo

En el diagrama se puede observar a modo de resumen, el procesamiento completo al cual se someten los pares de video original/degradado para obtener la estimación de la calidad percibida de video. En el *Anexo C*. Código fuente del algoritmo de Yonsei se puede observar el código fuente del algoritmo.

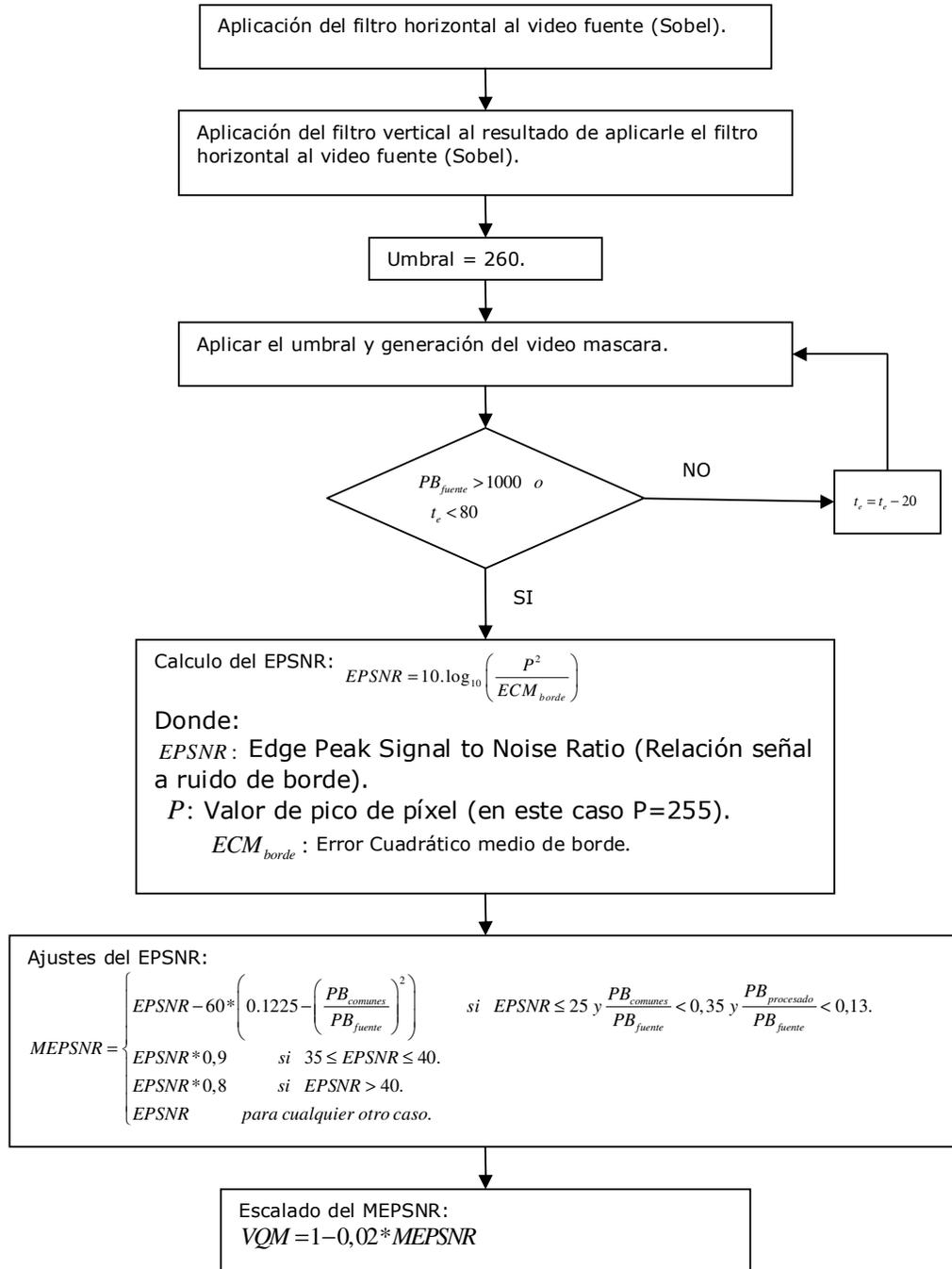


Figura 35. Diagrama del algoritmo de Yonsei

### 3.5.4.4. Validación del algoritmo

El algoritmo se generó siguiendo detalladamente los pasos descritos en [49], [1] y [21]. Los videos publicados, únicamente son los de la fase I [15], pero los resultados publicados del Algoritmo de Yonsei existen únicamente para la fase II [16], por lo que en la validación se utilizaron los videos de referencia que se encuentran en ambas fases del proyecto FRTV del VQEG.

En las figuras *Figura 36* y *Figura 37*, se ven representados los valores obtenidos por el Prof. Chulhee Lee y sus colaboradores contra los resultados obtenidos por el algoritmo desarrollado.

Tomándose en cuenta la simplicidad del algoritmo y el apego estricto a las recomendaciones de referencia del mismo en [1] y [21], se dio por válida la implementación del algoritmo a pesar de las diferencias obtenidas para las secuencias 1 y 4 en el formato NTSC (525 líneas).

Cabe mencionar que al pie de cada página en las tablas de resultados, tanto en [1] como en [21], respecto a los resultados obtenidos por el Prof. Chulhee Lee, se advierte que luego que el modelo fue presentado, se encontraron errores de operación y registro del algoritmo que fueron corregidos al momento de ser publicado en la norma<sup>9</sup>.

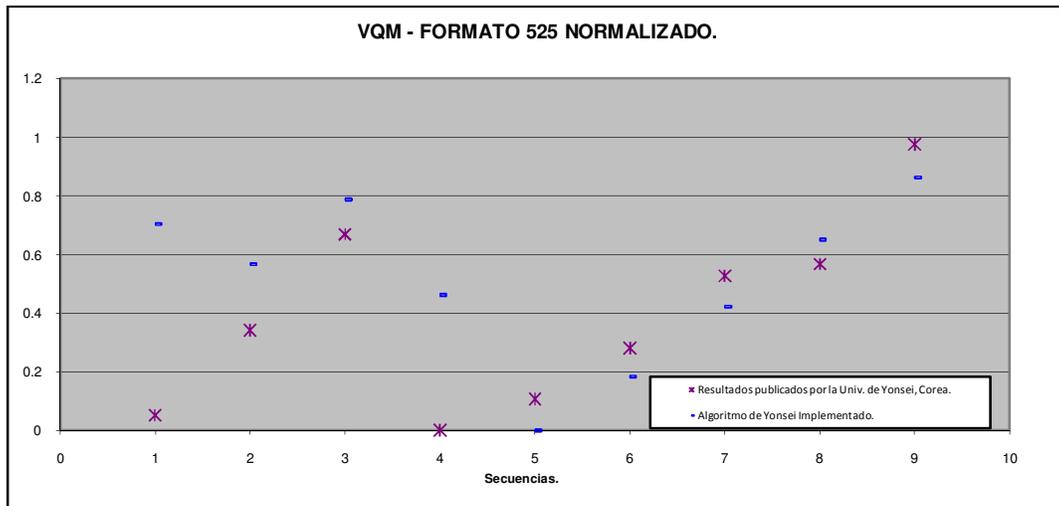


Figura 36. VQM - Formato NTSC normalizado

<sup>9</sup> La norma ITU-R BT.1683 [1] dice textualmente en inglés “After the model was submitted, registration and operator errors were found. The objective data presented in this Annex is the same data as in the VQEG Phase II Final Report. Consequently, when the method described in this Annex is properly implemented, the user may obtain different objective data from those of Table B.4.”,

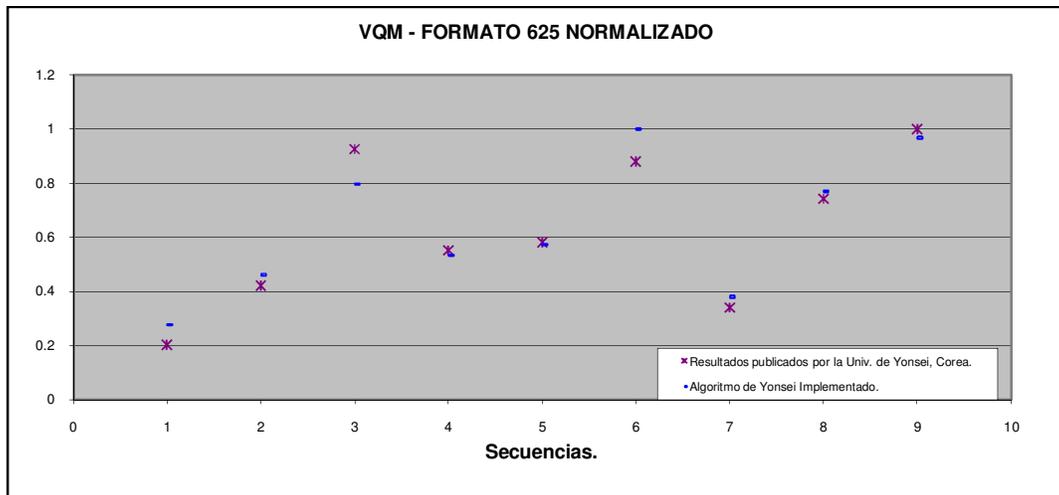


Figura 37. VQM – Formato PAL normalizado

Se puede observar que en las secuencias con formato 625, el algoritmo generado y los resultados del utilizado en la VQEG son prácticamente idénticos. En las secuencias con formato 525, se encontraron secuencias en las cuales hubo diferencias. El algoritmo se validó tomando en cuenta el resultado global.

### 3.6. Análisis estadístico

En este capítulo se describe la metodología utilizada para analizar los datos obtenidos tanto de las pruebas subjetivas como de las objetivas. Es en base a este análisis es que las conclusiones de este estudio se sustentan.

De la misma forma que para la realización de las pruebas subjetivas, el análisis estadístico de los datos se realiza siguiendo los lineamientos marcados por el VQEG, los cuales a su vez se basan en normas y recomendaciones aceptadas internacionalmente, particularmente la recomendación ITU-R BT.1676 [28].

La primera fase del análisis estadístico consiste en pre-procesar los datos, tanto los obtenidos por pruebas subjetivas como los generados por los algoritmos, de forma de que puedan ser llevados a una misma escala común para luego ser comparados.

#### 3.6.1. Normalización de datos subjetivos.

Como se indica en 0, el método de evaluación seleccionado para llevar adelante las pruebas subjetivas es DSCQC. Este arroja un par de calificaciones para cada secuencia de videos original-procesado, donde uno de los valores corresponde al video original y otro al video degradado, con los cuales es posible calcular la diferencia. Este resultado es luego normalizado de forma de llevar la puntuación a una escala entre 0 y 1 mediante la fórmula:

$$DMOS_{ij} = \frac{Dif\_Video_j - Min_i}{Max_i - Min_i}$$

Ecuación 11

Donde  $DMOS_{ij}$  es el puntaje diferencial del sujeto  $i$  para el video  $j$ ,  $Dif\_Video_j$  es la diferencia entre el valor original y el degradado para el video  $j$  al que se quiere calificar,  $Min_i$  es la puntuación diferencial mínima del sujeto  $i$  y  $Max_i$  es la puntuación diferencial máxima para el sujeto  $i$  [16].

#### 3.6.2. Eliminación de sujetos.

En toda muestra poblacional se encuestan sujetos cuyo comportamiento no responde a la media de la población, los cuales si no son identificados y tratados apropiadamente pueden llevar a resultados inexactos. Por este motivo se determinó si la muestra incluía "outliers", individuos cuyas calificaciones se alejan demasiado de la media poblacional en más del 5% de los valores, siguiendo la metodología empleada por el VQEG.

Para determinar qué valores pueden ser considerados outliers es necesario conocer la distribución de la muestra, ya que hay distribuciones más propensas a tener valores alejados de la media poblacional. En particular, es de interés determinar si las muestras se ajustan a una distribución normal o no. Existen varios test con los cuales es posible determinar si la distribución de una muestra es normal, como el test  $\beta_2$ , el cual fue utilizado en este estudio de acuerdo a la norma ITU BT-500-10 [27].

En los casos donde la distribución de probabilidades es normal, se consideran outliers los valores que se alejan más de 2 veces la desviación estándar, mientras que en el resto de los casos, para que el valor se considere un outlier, la distancia a la media tiene que ser mayor a  $\sqrt{20}$  veces la desviación estándar.

Luego, se re calcula la media y la desviación estándar del DMOS de cada video ( $\overline{DMOS_j}$ ) sin tener en cuenta los valores provenientes de los sujetos eliminados. Es importante resaltar que este proceso no es iterativo, es decir, una vez hecha la eliminación y re calculada la media y la desviación estándar, no se vuelve a eliminar sujetos.

El valor que se debe comparar con los valores obtenidos por los algoritmos es:

$$\overline{DMOS_j} = \sum_i \frac{DMOS_{ij}}{n} \quad i = 1 \dots n$$

Ecuación 12

Donde n es el número de sujetos que observaron el video j

### 3.6.3. Ajuste de los datos

Los puntajes arrojados por los algoritmos (*VQM*) deben estar correlacionados con el *DMOS* de los espectadores de forma predecible y repetible. La relación entre el *VQM* y *DMOS* no tiene porque ser lineal, ya que las calificaciones subjetivas se comportan en forma no lineal en los extremos del rango de prueba [16]. Para facilitar la comparación entre ambos indicadores se estima su relación mediante una regresión no lineal utilizando la siguiente función:

$$DMOS_p = \frac{\beta_1}{\left(1 + e^{(-\beta_2 \times (VQM - \beta_3))}\right)}$$

Ecuación 13

Con el fin de determinar los parámetros  $\beta_1$ ,  $\beta_2$  y  $\beta_3$ , se ajusto la función a los valores [*DMOS*, *VQM*] para cada uno de los conjuntos de valores *VQM* obtenidos mediante los algoritmos, obteniendo así respectivas funciones para cada algoritmo.

Una vez conocida la función, se utilizan los valores de *VQM* para obtener el  $DMOS_p$ , conjunto de valores pronosticados que son los comparados con el *DMOS* que se obtiene en las pruebas subjetivas.

### 3.6.4. Métricas para la evaluación de los algoritmos

La performance de los algoritmos debe ser evaluada con respecto a tres aspectos de la capacidad de estimar valores de *DMOS*.

1. La exactitud de la predicción
2. La monotonía de la predicción
3. La consistencia de la predicción.

En este sentido el VQEG plantea una serie de test a ejecutar para evaluar los algoritmos, cada uno con distinto foco dentro de los aspectos mencionados.

### Métrica 1 – Coeficiente de correlación lineal de Pearson

Este coeficiente indica si hay una relación lineal entre las variables *DMOS* y *DMOSp*. Es una métrica que evalúa la exactitud de la predicción. Cuanto más cercano a 1 es el resultado, las variables están más relacionadas.

La fórmula utilizada para calcular el coeficiente es:

$$r = \frac{\sigma_{XY}}{\sigma_X \cdot \sigma_Y}$$

Ecuación 14

Donde  $\sigma_{xy}$  representa la covarianza y  $\sigma_x$  y  $\sigma_y$  son las varianzas de las variables de interés.

### Métrica 2 – Coeficiente de correlación de Spearman

El test de Spearman ordena los datos del *DMOSp* y *DMOS* de mayor a menor y calcula el coeficiente en función a la posición de cada dato.

Si tenemos un conjunto de medidas objetivas ( $O_i$ ) y sus respectivas observaciones subjetivas ( $S_i$ ), y creamos a partir de estos conjuntos dos conjuntos,  $o_i$  y  $s_i$ , que contienen el orden de los valores de  $O_i$  (al mayor valor de  $O_i$  se le asigna el valor 1, al siguiente valor el 2 y sucesivamente hasta que al menor valor se le asigna el valor N) y  $S_i$  respectivamente. Si llamamos  $d_i$  a la diferencia  $d_i = o_i - s_i$ , el coeficiente de Spearman se calcula como:

$$\rho = 1 - \frac{6 \sum_{i=1}^N d_i^2}{N(N-1)}$$

Ecuación 15

Es particularmente útil para evaluar la monotonía de la predicción y al igual que el test de Pearson, cuando más cercano a 1 es el resultado, la correlación es mayor.

### Métrica 3 – Cociente de Outliers

Se entiende por outliers a los valores del *DMOSp* que se alejan demasiado del valor *DMOS*. Es evidente que cuanto menor sea el cociente de estos valores, el algoritmo se comporta mejor.

Se considera un punto outlier cuando:

$$|DMOSp - DMOS| \geq 2 \times DMOS\_std\_error$$

Ecuación 16

Donde *DMOS\_std\_error* es el error estándar del *DMOS* ( $= \sigma/\sqrt{N}$ )

Las métricas 4, 5 y 6 fueron obtenidas de la recomendación ITU-R BT.1676 [28]. En este documento se especifica una metodología para comparar métricas de calidad de video, que incluye el pre-procesamiento para llevar las métricas a una

escala común, donde puedan ser comparadas, el ajuste de los valores a una función polinomial para eliminar características indeseables y se detallan las métricas propuestas.

También se incluye un código MATLAB descrito en el *Anexo D. Código fuente para el cálculo de las Métricas*, que realiza un ajuste polinomial de los datos y posteriormente calcula las métricas detalladas a continuación.

#### **Métrica 4 – Potencia de resolución**

Dadas dos medidas objetivas de degradación provenientes de dos videos distintos, es necesario saber cuál debe ser la diferencia mínima entre ellas, para poder afirmar que la degradación real es distinta en ambos videos. Esta métrica busca cuantificar cual es esta diferencia.

El algoritmo que evalúa esta métrica, toma todas las posibles combinaciones de pares de degradaciones y evalúa mediante un test Z, aplicado a la diferencia en las puntuaciones subjetivas, la probabilidad de que las muestras provengan de distribuciones con medias distintas.

La fórmula para calcular el valor de Z es la siguiente:

$$Z_{ij} = \frac{S_i - S_j}{\sqrt{\frac{V_i}{N_i} + \frac{V_j}{N_j}}}$$

**Ecuación 17**

Donde  $V_i$ ,  $V_j$  son las varianzas de los videos i y j respectivamente

$N_i$ ,  $N_j$  son la cantidad de observaciones de cada video.

Para obtener la probabilidad asociada a cada valor de Z se utiliza la distribución de probabilidad normal.

$$p(Z) = cdf(Z) = (2\pi)^{-1/2} \int_{-\infty}^Z e^{-z^2/2} dZ$$

**Ecuación 18**

Esta probabilidad, que como se explico es calculada con la diferencia subjetiva  $S_i-S_j$ , tiene asociada también una diferencia objetiva  $O_i-O_j$ . De esta forma podemos generar una tabla en la que a cada diferencia  $O_i-O_j$  le corresponde una probabilidad o nivel de significancia.

Si bien la relación entre  $O_i-O_j$  y el nivel de significancia tienen que tener cierta correlación, esta relación no es necesariamente monótona (dado que la relación entre  $O_i$  y  $S_i$  tampoco es monótona). Para mitigar esta falta de monotonía, se realiza un promedio por intervalos, dividiendo todo el rango de diferencias  $O_i-O_j$  en M intervalos iguales. Luego, se calcula el promedio del nivel de significancia para cada intervalo.

Por último, dado el nivel de significancia requerido (valores comunes son 0.68, 0.75, 0.90 y 0.95), se determina a través de una interpolación la mínima diferencia que deben tener dos puntuaciones objetivas  $O_i-O_j$  para poder afirmar que una degradación es mayor que la otra con el nivel de significancia.

El valor arrojado por esta métrica se puede interpretar como la apreciación el algoritmo.

### Métrica 5 – Raíz del error cuadrático medio

Esta métrica, probablemente la más conocida de todas, es simplemente una medida del error total cometido. Si bien es útil para saber la exactitud de un algoritmo, no nos da información sobre la consistencia del mismo.

La fórmula para calcularla es la siguiente:

$$RMSE = \sqrt{\frac{1}{N-D} \sum_{i=1}^N (O_i - S_i)^2}$$

Ecuación 19

Donde:

$N$  es el número total de pares de videos a evaluar

$D$  es el número de grados de libertad del polinomio con el que se ajustaron los valores

$O_i$  es el valor del VQM del video  $i$

$S_i$  es el valor del DMOS del video  $i$

### Métrica 6 – Errores de clasificación

Al igual que la métrica 3, los errores de clasificación brindan una idea de la consistencia de la medida. Dadas dos degradaciones provenientes de dos videos cualesquiera, se considera un error de clasificación cuando las calificaciones subjetivas y las arrojadas por los algoritmos llevan a conclusiones distintas, en cuanto a la calidad de cada video en relación al otro.

Por ejemplo, si dados dos videos  $i$  y  $j$ , se tienen calificaciones subjetivas  $S_i$  y  $S_j$ , y las calificaciones objetivas  $O_i$  y  $O_j$ , puede suceder que  $S_i > S_j$  y  $O_i < O_j$ , lo cual lleva a una clasificación errónea. Esta métrica estima el porcentaje de estos errores de clasificación en cada algoritmo.

Realizando el mismo procedimiento que en la métrica 4, se estima el mínimo valor de  $S_i - S_j$  que nos garantiza que los videos tienen degradaciones diferentes con un cierto nivel de significancia, llamemos a este umbral  $U_s$ . Este valor es un parámetro fijo de la prueba.

Se puede definir también un umbral de clasificación para los valores objetivos, que llamaremos  $U_o$ . De esta forma, dadas dos degradaciones provenientes videos distintos, de acuerdo a la diferencia de los puntajes objetivos  $S_i$  y  $S_j$  se pueden clasificar en 3 regiones:

1. Si  $S_i - S_j > U_s$  entonces decimos que  $i$  es peor que  $j$
2. Si  $S_i - S_j < U_s$  decimos que  $i$  es equivalente a  $j$
3. Si  $S_i - S_j < -U_s$  decimos que  $i$  es mejor que  $j$

Llamemos estas regiones  $P_s$ ,  $I_s$  y  $M_s$  (Peor, Igual, Mejor respectivamente). Podemos hacer la misma clasificación tomando en cuenta las puntuaciones objetivas, y un umbral de decisión para los valores objetivos, el cual es variable. Llamemos también a estas regiones  $P_o$ ,  $I_o$  y  $M_o$

En este momento, cada par de puntuaciones puede ser clasificado en cada una de las regiones, lo que nos deja 9 posibles situaciones, ilustradas en la siguiente figura.

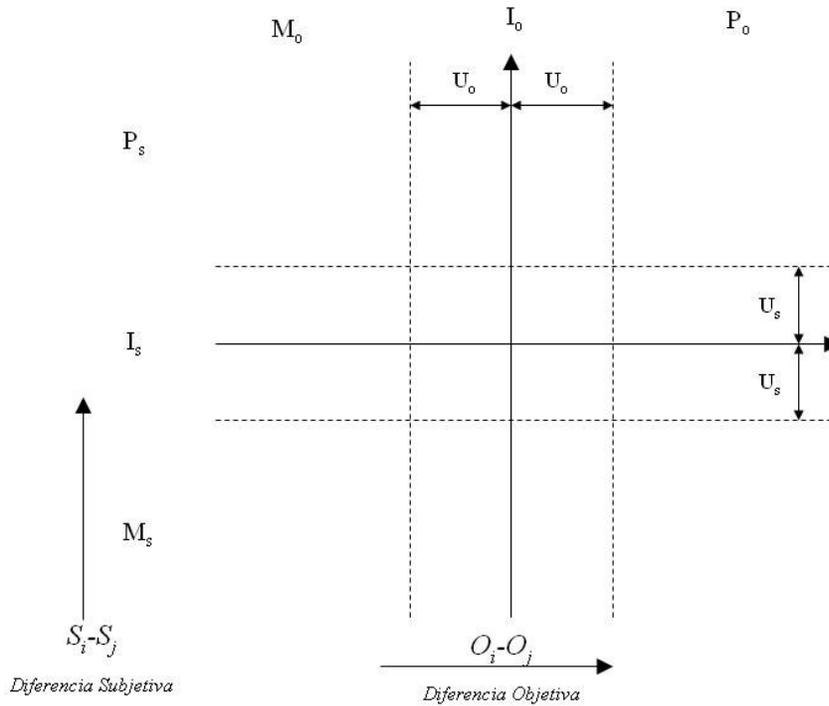


Figura 38. Diagrama de errores de clasificación

Así, cada punto pertenecerá a uno de los 9 cuadrantes definido por los umbrales. De estos cuadrantes, 3 representan una decisión correcta y 6 representan errores de clasificación de 3 tipos, falsos empates, falsas diferenciaciones y falsas clasificaciones. La *Figura 39* muestra como se ubican estos errores.

Falsa Clasificación	Falso Empate	Decisión Correcta
Falsa Diferenciación	Decisión Correcta	Falsa Diferenciación
Decisión Correcta	Falso Empate	Falsa Clasificación

Figura 39. Calcificación de los cuadrantes

El algoritmo luego, busca los valores de  $U_0$  mayores que el valor encontrado en la métrica 4 (potencia de resolución) que minimizan el error cometido. Si bien se encuentra un valor de  $U_0$  y su error asociado, la métrica solo toma en cuenta la magnitud del error. En nuestro caso, solo se considera el porcentaje de errores de clasificación, sin tener en cuenta que algún error puede ser más grave que otro (en particular, la falsa clasificación es el más grave y el falso empate el menos grave). Otra opción podría ser definir como métrica la suma ponderada de los distintos errores.

### Métrica 7 – Test F

Esta métrica, la cual no estaba inicialmente en el plan de pruebas de la VQEG, compara la performance del algoritmo respecto a un algoritmo ideal, en términos del error cuadrático medio respecto al puntaje otorgado por cada sujeto. El modelo ideal es aquel cuyas predicciones coinciden con valor promedio de los puntajes subjetivos, y por lo tanto la suma de los errores cuadráticos de cada puntuación respecto al promedio es mínima.

Es la única métrica que utiliza explícitamente los valores de degradación individuales.

El cálculo aplicado corresponde a la realización de un Test F

La fórmula para el cálculo es la siguiente:

$$F = \frac{MSE_{-vqm}}{MSE_{-opt.}} = \frac{\sum_{i=1}^N \sum_{j=1}^M (O_i - S_{ij})^2}{\sum_{i=1}^N \sum_{j=1}^M (\hat{S}_i - S_{ij})^2}$$

Ecuación 20

Donde

$O_i$  es el valor de *VQM* arrojado por el algoritmo para cada video *i*

$S_{ij}$  es el valor de *DMOS* del sujeto *j* para el video *i*

$\hat{S}_i$  es el valor de *DMOS* promedio para el video *i*

Resulta evidente que cuanto *F* se aproxima más a 1 el modelo es mejor. En caso que se estudia en este proyecto, para aceptar la hipótesis nula del test (esto es que ambos modelos son equivalentes) *F* debe ser menor a 1,13 [50].

## Capítulo 4. Resultados obtenidos

En el *Anexo E. Tablas de resultados de los test subjetivos*, se pueden observar los resultados de las pruebas subjetivas previo a cualquier procesamiento. Estos datos deben pasar por los procesos descritos en 3.6.1 y 3.6.2 para poder ser comparado luego con los algoritmos.

Como resultado del proceso de eliminación de sujetos se descartaron las puntuaciones provenientes del sujeto nº 7, quien no respondía de acuerdo a la media poblacional en un 7% de las observaciones, siendo el máximo permitido un 5%.

El valor que se tomó como DMOS de cada video es el promedio de los valores diferenciales de cada individuo. La desviación estándar obtenida es levemente mayor a la de la fase 1 del VQEG, en la cual se utilizaron los mismos videos de referencia. Como se menciona en 3.4.3, la desviación estándar promedio hallada para los valores surgidos de la fase I del proyecto FRTV del VQEG son de 0,1588 para los videos en formato NTSC, y de 0,1719 para los videos en formato PAL, mientras que los valores surgidos de este estudio son de 0,1748 para el formato NTSC y de 0,1458 para el formato PAL.

En cuanto a los test objetivos, los resultados se pueden observar en el *Anexo F. Tablas de resultados de los test objetivos*. Durante el proceso de evaluar los pares de videos original-procesado, fue necesario dejar de lado los pares en los cuales no había diferencias, debido a que tanto el PSNR como el algoritmo de la Universidad de Yonsei, son incapaces de lidiar con estos videos. Esto se debe a que el MSE es 0 y el VQM resultaría infinito (ver definición de PSNR en 3.5.3). De todas formas, los algoritmos de la NTIA y el CPqD, arrojan un VQM de 0 cuando los no hay degradación alguna, tal como es de esperar.

El primer paso del procesamiento de las pruebas subjetivas, es un ajuste de los valores arrojados por los algoritmos (VQM) a los datos del DMOS, obtenidos a través de las pruebas subjetivas. Las características deseables de la función de ajuste son que sea monótona, que este definida en todo el intervalo de variación del VQM y que su codominio para el intervalo del VQM sea el intervalo [0,1], siendo 0 para el VQM de dos videos iguales y 1 para la máxima degradación.

La función con la que se ajustaron estos datos es de la forma:

$$DMOS_p = \frac{\beta_1}{\left(1 + e^{(-\beta_2 \times (VQM - \beta_3))}\right)}$$

Ecuación 21

Donde  $\beta_1, \beta_2$  y  $\beta_3$  son los parámetros a determinar.

Esta función cumple con la característica de monotonía buscada, y está definida para todos los reales.

Una vez determinados los valores beta de la función, se utiliza la misma para calcular los valores de DMOS<sub>p</sub>, los cuales serán comparados con los valores del DMOS.

A continuación se presentan las graficas de dispersión de los valores del VQM y DMOS y la función resultante del ajuste.

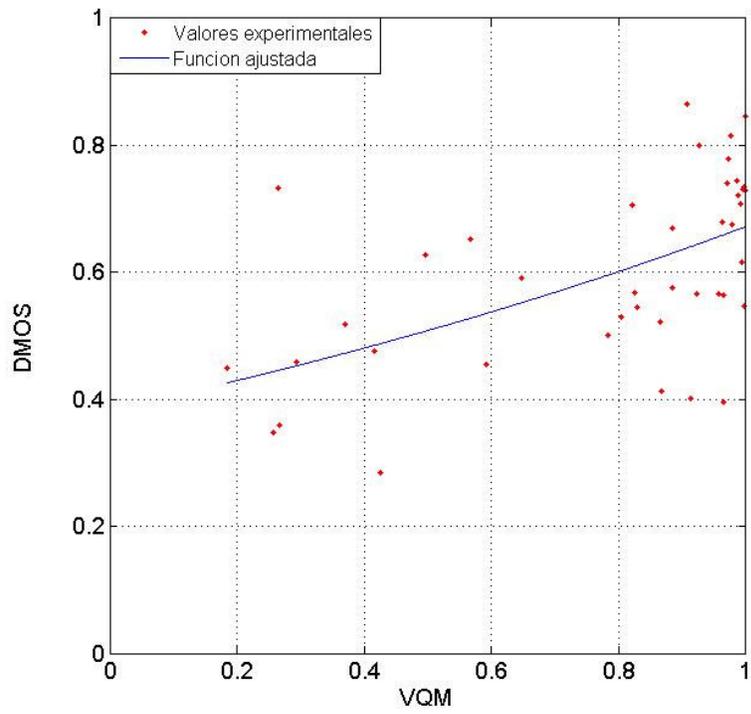


Figura 40. Dispersión del DMOS y el VQM obtenido del algoritmo del CPqD para el formato PAL

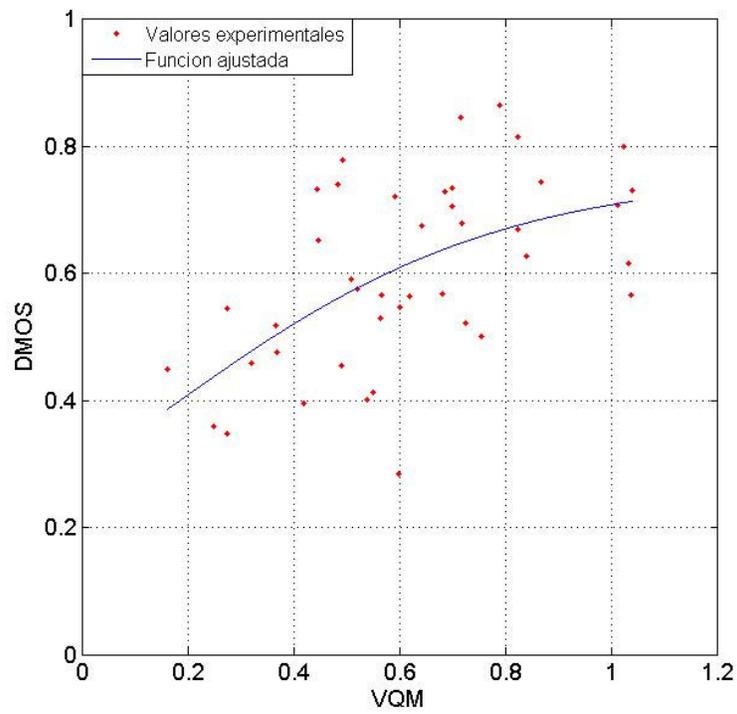


Figura 41. Dispersión del DMOS y el VQM obtenido del algoritmo de la NTIA para el formato PAL

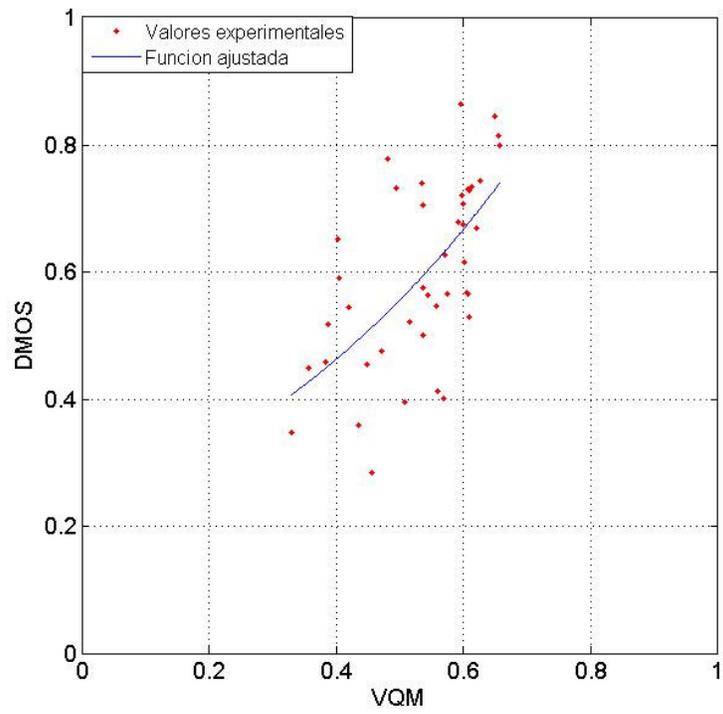


Figura 42. Dispersión del DMOS y el VQM obtenido del algoritmo de Yonsei para el formato PAL

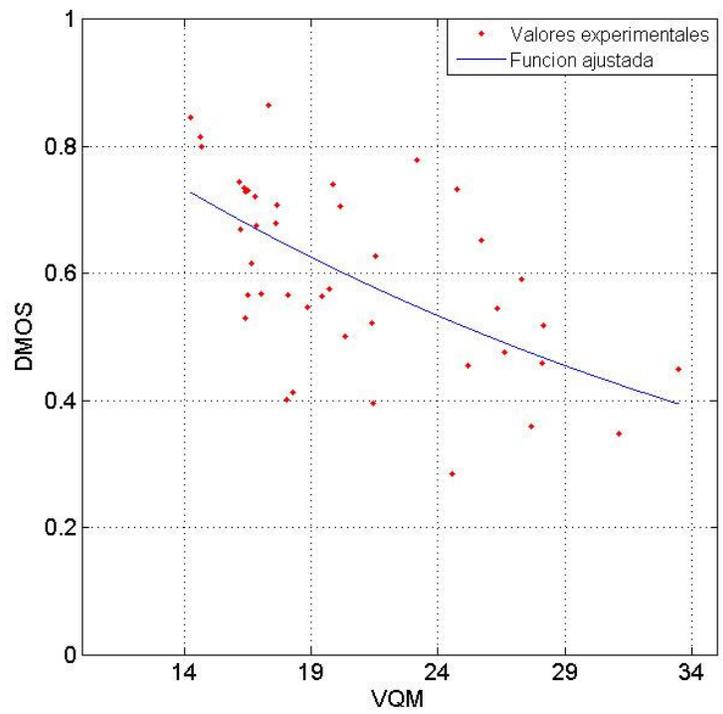


Figura 43. Dispersión del DMOS y el VQM obtenido del PSNR para el formato PAL

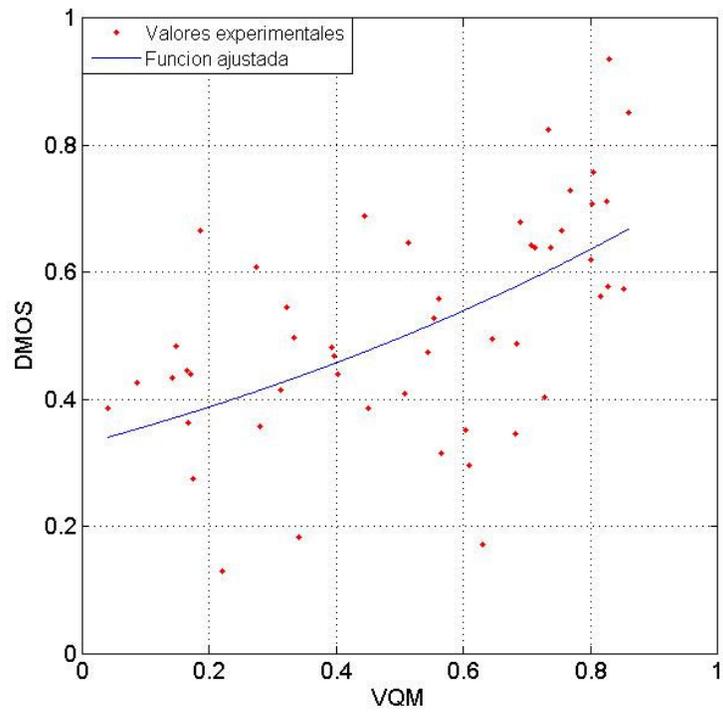


Figura 44. Dispersión del DMOS y el VQM obtenido del algoritmo del CPqD para el formato NTSC

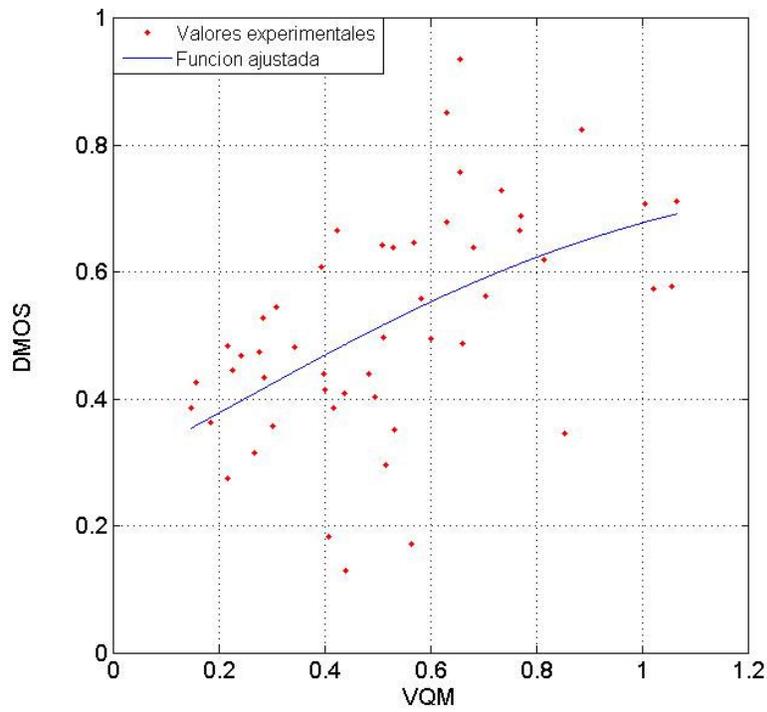


Figura 45. Dispersión del DMOS y el VQM obtenido del algoritmo de la NTIA para el formato NTSC

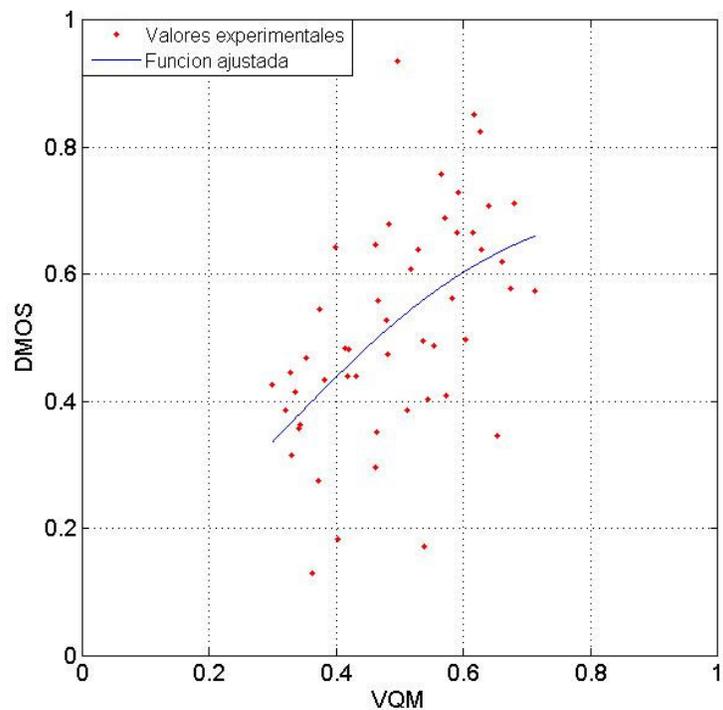


Figura 46. Dispersión del DMOS y el VQM obtenido del algoritmo de Yonsei para el formato NTSC

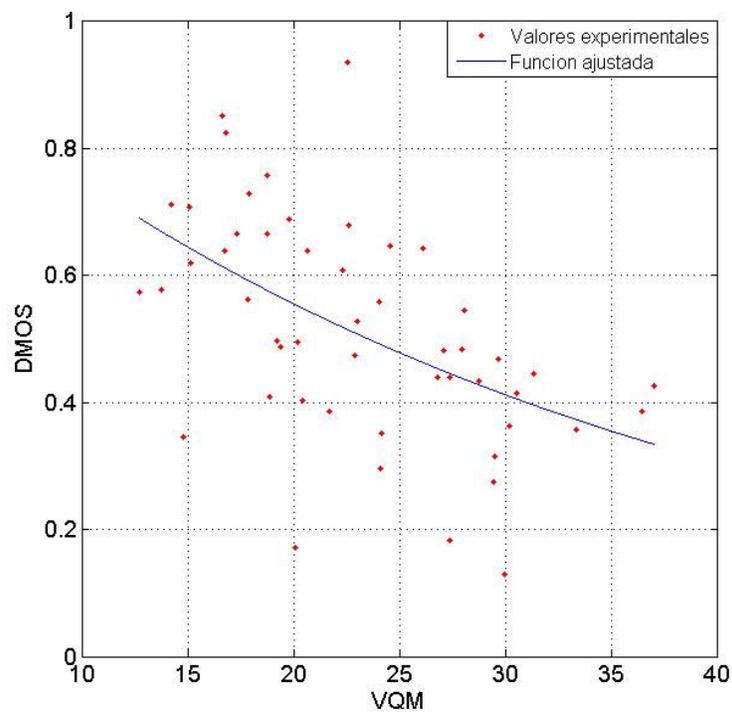


Figura 47. Dispersión del DMOS y el VQM obtenido del PSNR para el formato NTSC

Se puede ver que las funciones encontradas no poseen un rango como el deseado, ya que, para el intervalo de variación de los VQM, el intervalo de valores en el codominio va desde 0,3 a 0,8.

Una vez que se conocen los valores  $\beta_i$  beta de la función para cada algoritmo, utiliza la misma para calcular los valores de DMOSp, los cuales serán comparados con los valores del DMOS.

Para evaluar las métricas propuestas por el VQEG se implementó una rutina en MATLAB (cuyo código se puede ver en el *Anexo D. Código fuente para el cálculo de las Métricas*) que toma como entrada los valores diferenciales de cada individuo para cada video, el VQM arrojado por el algoritmo a evaluar y la relación entre los sentidos crecientes del DMOS y el VQM. Este último parámetro se debe a que en el PSNR la relación es inversa, lo cual se puede ver por la pendiente negativa en los gráficos anteriores.

La rutina de MATLAB realiza el ajuste descrito anteriormente, calcula el DMOS y su desviación estándar, así como el DMOSp la desviación estándar correspondiente, para luego calcular cada una de las métricas propuestas por el VQEG

Fue necesario cambiar el valor del nivel del significancia para la métrica 4, potencia de resolución, dado que no todos los algoritmos llegaban a un nivel de significancia del 95%. De las opciones usuales de este valor (68%, 75%, 90%, 95%) se tomó la del 75% ya que todos los algoritmos llegan a este valor. Esto no afecta la validez del estudio, ya que todos los algoritmos son comparados contra un mismo umbral, si bien hace que este valor no pueda ser comparado con la misma métrica que en el estudio del VQEG fase 2.

La métrica 7 tiene como hipótesis que los residuos  $O_i - S_{ij}$  deben tener una distribución normal. Para comprobar esta hipótesis se utilizó un test de Shapiro-Wilks sobre los residuos. Se encontró que tomando todos los residuos individuales ( $O_i - S_{ij}$ , 756 puntos en el experimento de 525 y 644 en el de 625 ) no es posible afirmar que ninguno de los algoritmos tenga residuos normales, aunque los valores obtenidos del test se aproximan a los de una distribución normal. Al aplicar este mismo test a los residuos del DMOS  $O_i - \bar{S}_i$ , se encontró que los residuos tienen las características de una distribución normal. Estos resultados son coherentes con los obtenidos por el VQEG en su experimento [16], y por lo tanto, podemos asumir que los resultados del test F tienen la misma validez que en dicho experimento.

A continuación se presenta las tablas con los resultados de las 7 métricas tanto para la prueba en formato PAL como la formato NTSC.

PAL	Yonsei	NTIA	CPqD	PSNR
Correlacion Pearson	<b>0.6177</b>	0.5619	0.5485	0.5953
Correlacion Spearman	<b>0.6179</b>	0.5396	0.5962	0.6224
Cociente Outliers	<b>0.3810</b>	0.4286	0.4048	<b>0.3810</b>
Resolving Power	<b>0.1501</b>	0.1661	0.1610	0.1666
RMSE	<b>0.1176</b>	0.1236	0.1250	0.1201
Classification Errors	<b>0.5075</b>	0.5261	0.5226	0.5168
MSE model /MSE optim	<b>1.4181</b>	1.4559	1.4346	1.4314

Tabla 9. Resultados de las métricas para el formato PAL

NTSC	Yonsei	NTIA	CPqD	PSNR
Correlacion Pearson	0.5349	0.5290	<b>0.5569</b>	0.5204
Correlacion Spearman	0.5769	0.5818	0.5727	<b>0.5821</b>
Cociente Outliers	<b>0.3800</b>	<b>0.3800</b>	0.4000	0.4000
Resolving Power	<b>0.1378</b>	0.1405	0.1901	0.1465
RMSE	0.1508	0.1515	<b>0.1483</b>	0.1524
Classification Errors	0.4988	<b>0.4906</b>	0.5184	0.5151
MSE model /MSE optim	1.5354	1.5380	<b>1.5177</b>	1.5463

Tabla 10. Resultados de las métricas para el formato NTSC

En la prueba en formato PAL, el algoritmo con mejor respuesta es el de la Universidad de Yonsei, el cual supera a los demás algoritmos en las 7 métricas calculadas (PSNR tiene el mismo cociente de outliers). En la prueba en formato NTSC, los resultados son mas dispersos, lo que no permite concluir que un algoritmo se desempeña mejor que otro.

Todos los números reflejan resultados lejanos a los deseados y a los obtenidos por la fase 2 del VQEG. Incluso el PSNR, algoritmo que no estamos evaluando sino que tomamos como referente, muestra un considerable deterioro en los valores de las métricas.

Las correlaciones de Pearson y Spearman, con valores entre 0,52 y 0,62, se ubican en las franjas consideradas de correlación media. El cociente de Outliers es muy grande, llegando al 43%. Otros valores, como el RMSE, no tienen un significado por sí mismos, solo nos permiten comparar un algoritmo con otro. En este sentido, todos los algoritmos tienen valores del mismo orden y no es posible destacar uno sobre los demás.

Resulta necesario hacer un análisis de algunos valores cuya interpretación no es inmediata. La métrica N° 6, como fue explicado en 3.6.4, busca medir la cantidad de errores que se cometen al comparar la calidad percibida de dos videos distintos a través de los VQM arrojados por un algoritmo. Más precisamente, se realiza una clasificación a través de los valores de DMOS, el cual refleja la opinión de la media poblacional (valor "real"). Este proceso clasifica la calidad de dos videos como equivalente si la diferencia del DMOS es menor que un umbral, o como una calidad superior a la otra cuando la diferencia del DMOS supera dicho umbral, obteniendo 3 posibles regiones (calidad A mayor que calidad B, ambas calidades equivalentes y calidad A menor que calidad B). Luego, utilizando los valores del VQM en lugar del DMOS, se hace la misma clasificación. Decimos que hay un error de clasificación cuando la clasificación realizada a través del VQM no coincide con la realizada utilizando el DMOS. En este caso, esta métrica oscila entre 49% y 53%, lo que constituye números no muy esperanzadores, teniendo en cuenta que los mismos algoritmos en un experimento sin pérdidas de paquetes lograron errores de clasificación menores al 20%.

Un análisis similar puede hacerse para el valor del resolving power. El valor hallado corresponde a una diferencia en el DMOSp que garantiza que un video tiene mayor calidad de imagen que otro, con un nivel de significancia del 75%. O dicho de otra forma, si decimos que los DMOS de dos videos son distintos siempre que tengan una diferencia en el DMOSp mayor al resolving power, cometeríamos un error en el 25% de las ocasiones. Si suponemos un VQM que no tenga ninguna relación con el DMOS, por ejemplo, que genere el VQM aleatoriamente, la clasificación de un DMOS como mayor que otro también será aleatoria, y las chances de que la clasificación sea correcta serán del 50%, por lo que un aumento

de un 25% en este valor no parece un avance tan importante, y los valores del resolving power obtenidos, que van de 0,14 a 0,19 parecen muy altos para la confianza que nos otorgan.

En cuanto al cociente de MSE, estos valores están lejos de 1,13 y 1,14, para las pruebas en formato PAL y NTSC respectivamente, que permitirían concluir que el DMOS y un algoritmo son estadísticamente iguales.

Adicionalmente, se calcularon los indicadores F del test F para la comparación entre los distintos algoritmos. Las siguientes tablas muestran los resultados.

PAL	NTIA	CPqD	PSNR
Yonsei	1.0267	1.0116	1.0094
PSNR	1.0171	1.0022	
CPqD	1.0149		

Tabla 11. Comparación de la varianza residual de los distintos algoritmos para el formato PAL

NTSC	PSNR	NTIA	Yonsei
CPqD	1.0189	1.0134	1.0117
Yonsei	1.0071	1.0017	
NTIA	1.0054		

Tabla 12. Comparación de la varianza residual de los distintos algoritmos para el formato NTSC

En estas tablas se ven los cocientes de los errores cuadráticos de los valores individuales respecto al DMOS<sub>p</sub> de dos algoritmos, como indica la siguiente ecuación.

$$F = \frac{MSE_{-vqm_1}}{MSE_{-vqm_2}} = \frac{\sum_{i=1}^N \sum_{j=1}^M (O^1_i - S_{ij})^2}{\sum_{i=1}^N \sum_{j=1}^M (O^2_i - S_{ij})^2}$$

Ecuación 22

Nuevamente nuestros umbrales son de 1,13 y 1,14, de acuerdo a este test todos los algoritmos son estadísticamente equivalentes, con un nivel de confianza del 95%.

A continuación se muestran las graficas de dispersión del DMOS y el DMOS<sub>p</sub>, en la que los puntos que son Outliers, de acuerdo a la definición de la métrica 3, se ven en rojo.

La concentración de los puntos sobre el centro en los gráficos se debe al ajuste realizado por los algoritmos. Como fue mencionado, el rango del DMOS<sub>p</sub> está limitado al rango de la función de ajuste.

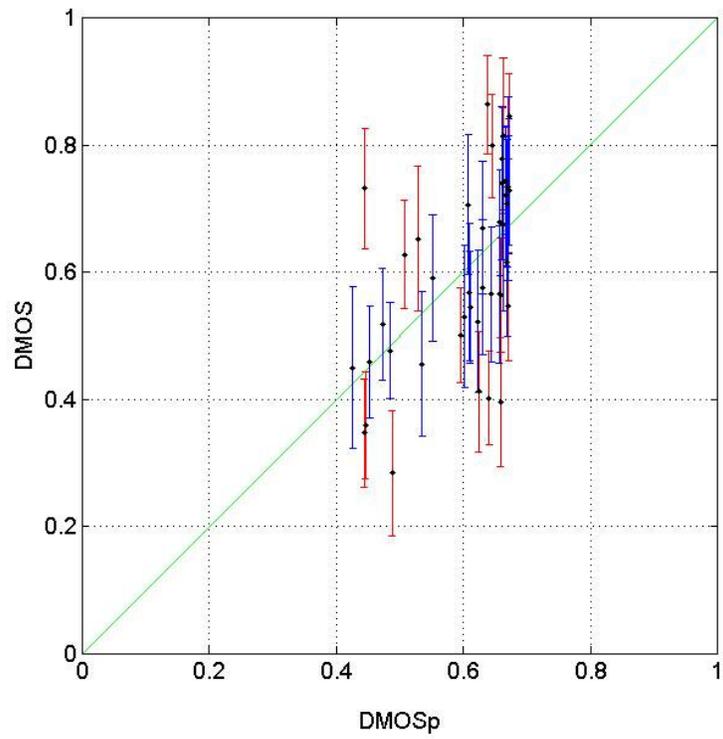


Figura 48. Dispersión del DMOS y el DMOSp obtenido del algoritmo del CPqD para el formato PAL

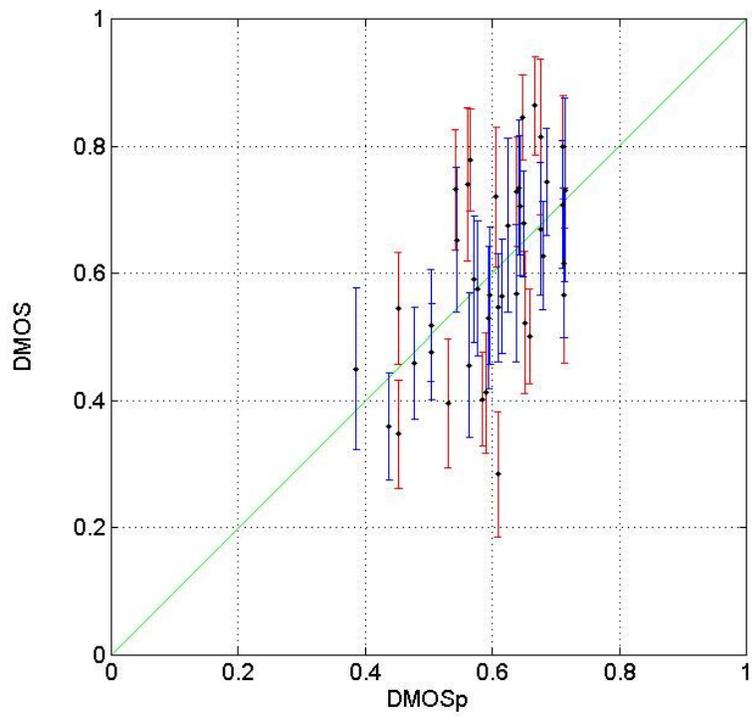


Figura 49. Dispersión del DMOS y el DMOSp obtenido del algoritmo de la NTIA para el formato PAL

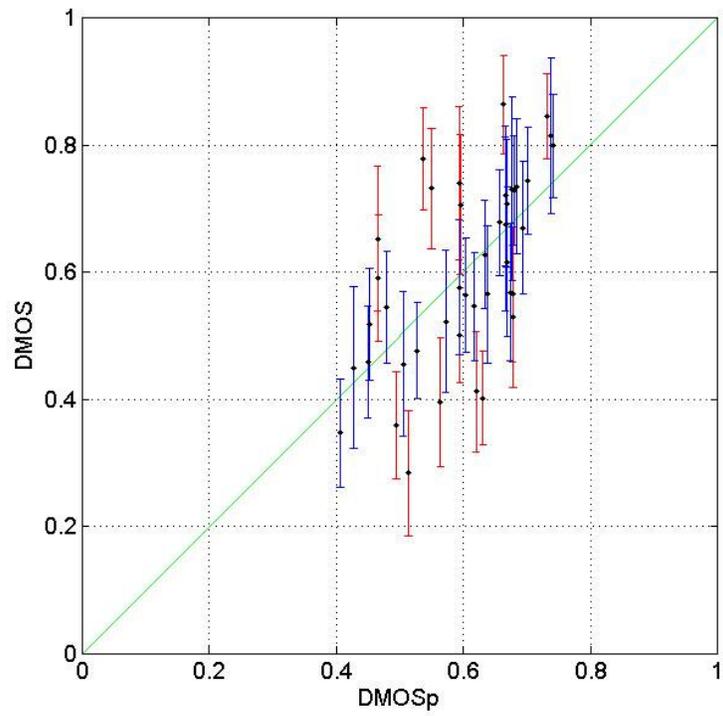


Figura 50. Dispersión del DMOS y el DMOSp obtenido del algoritmo de Yonsei para el formato PAL

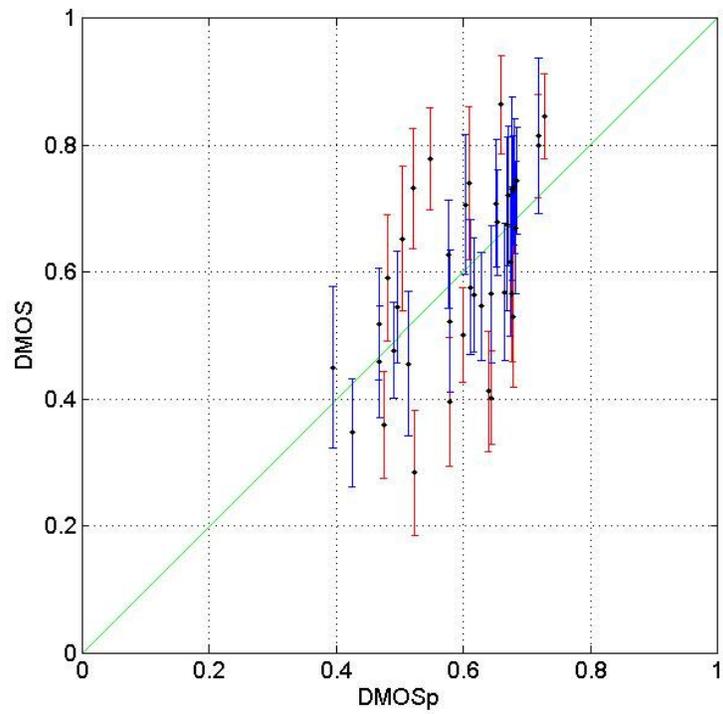


Figura 51. Dispersión del DMOS y el DMOSp obtenido del PSNR para el formato PAL

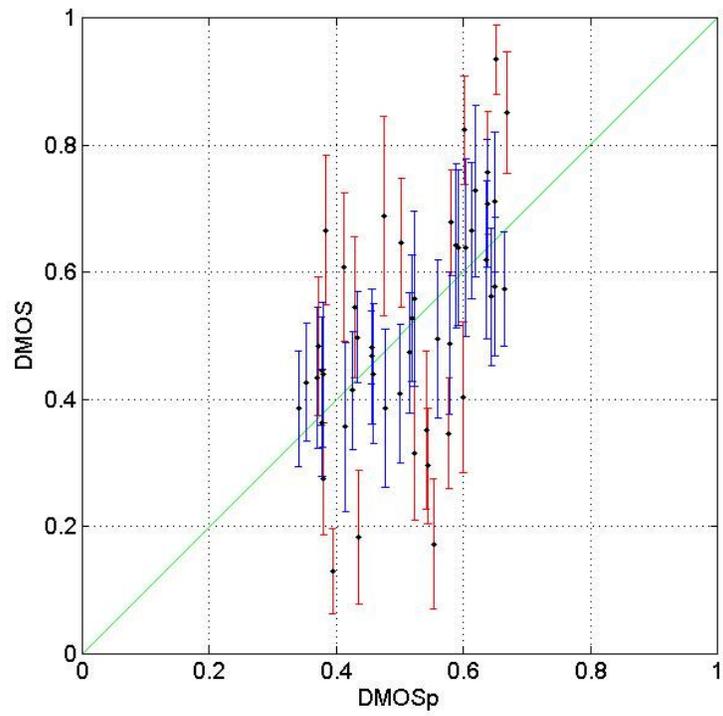


Figura 52. Dispersión del DMOS y el DMOSp obtenido del algoritmo del CPqD para el formato NTSC

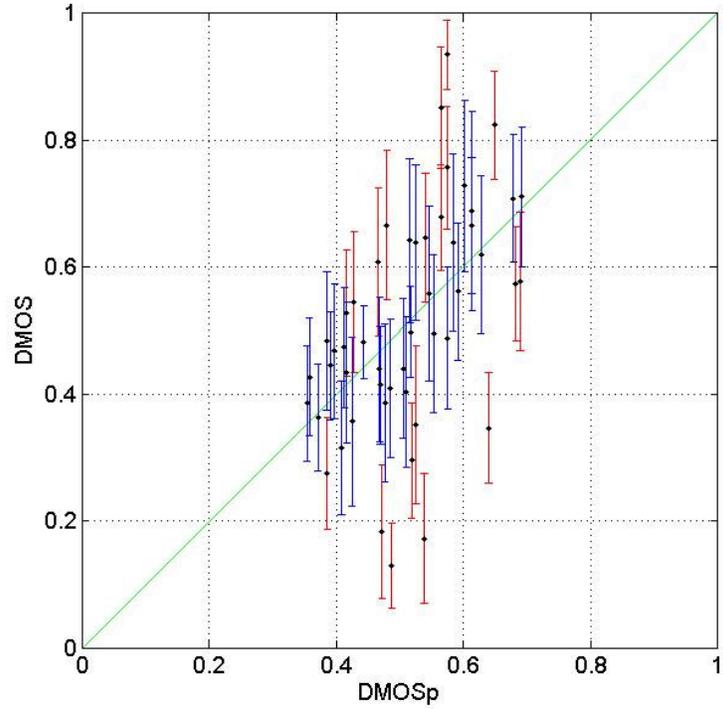


Figura 53. Dispersión del DMOS y el DMOSp obtenido del algoritmo de la NTIA para el formato NTSC

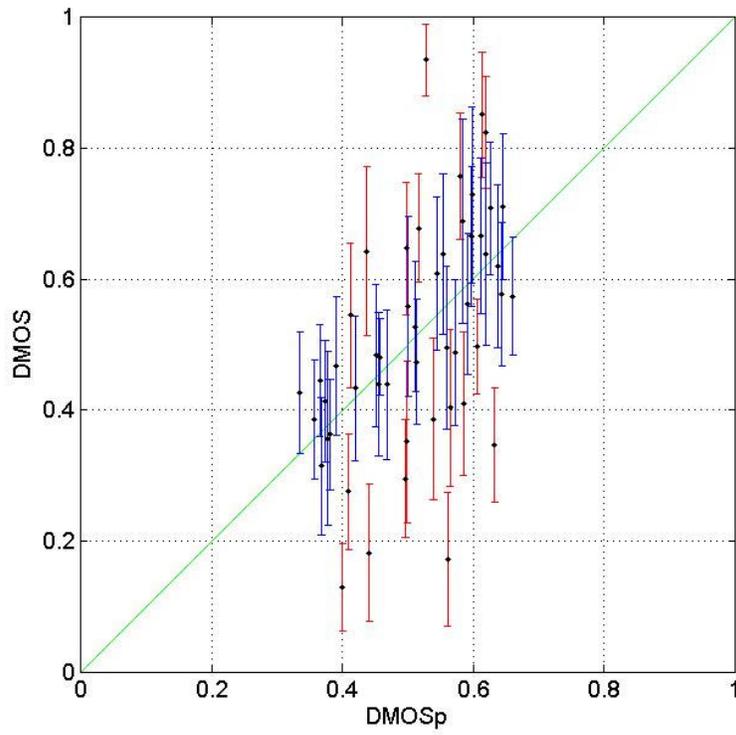


Figura 54. Dispersión del DMOS y el DMOSp obtenido del algoritmo de Yonsei para el formato NTSC

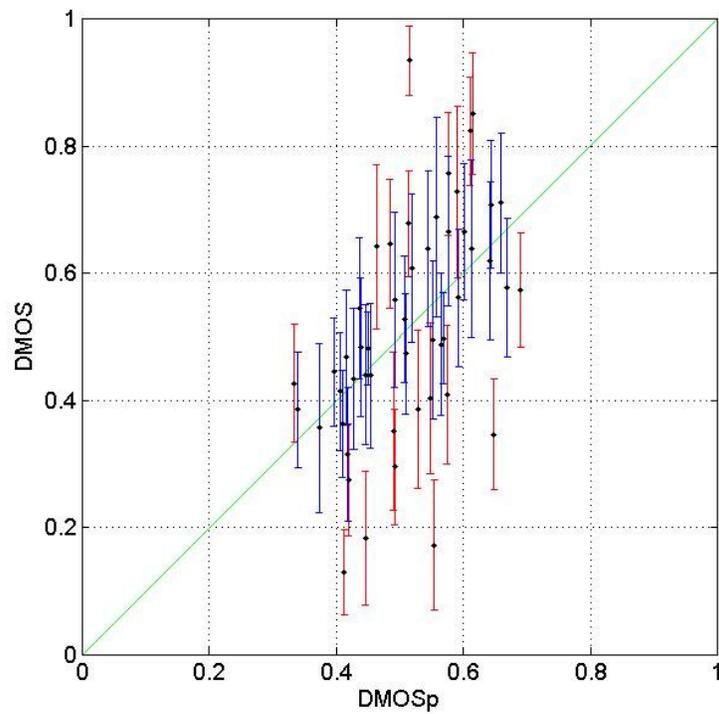


Figura 55. Dispersión del DMOS y el DMOSp obtenido del PSNR para el formato NTSC

## Capítulo 5. Conclusiones

De acuerdo a lo expuesto, se cuentan con suficientes elementos como para asegurar que todos los algoritmos estudiados son estadísticamente equivalentes entre sí y estadísticamente distintos del DMOS. Son de particular importancia para llegar a esta conclusión los valores de correlación lineal obtenidos, así como el resultado de comparar los algoritmos a través del test F (Tabla 11 y Tabla 12).

No solo son los algoritmos estadísticamente distintos al DMOS, sino que al ser estadísticamente equivalentes al PSNR no representan ningún avance en pos de la evaluación de la calidad de video percibida en un ambiente con degradaciones por pérdidas de paquetes.

Investigando las causas del bajo desempeño de los algoritmos, se encontró que durante el proceso de transmisión de las secuencias se dan pérdidas de cuadros completos, además de los artefactos visibles. Las pérdidas observadas, oscilaron entre 2 y 8 cuadros para videos de 200 y 240 cuadros. Este comportamiento no se puede controlar si únicamente se controla el porcentaje de probabilidad de pérdida de paquetes como es el caso de este estudio. Si se utiliza como referencia el modelo de capas TCP/IP, se puede decir que la pérdida de cuadros es una pérdida en capa de aplicación mientras que la pérdida de paquetes se produce claramente en la capa de red. La correlación entre la pérdida de paquetes y la de cuadros es baja. La relación entre estas dos pérdidas depende de varios factores como ser el codec de los videos, la tasa de compresión y el reproductor utilizado [51].

Inicialmente el universo de degradaciones aceptado para la 2da fase del VQEG incluía la posibilidad de que se perdieran o se repitieran cuadros, pero esta condición finalmente no fue testeada por dicho grupo, y los algoritmos aquí estudiados, que surgieron de la fase 2 del VQEG, no implementaban ningún control sobre pérdida de cuadros.

Algunos de estos algoritmos contemplan la posibilidad de corrimiento temporal e incluso contemplan corrimiento espacial, pero no esperan pérdidas de cuadros, y los corrimientos máximos esperados son de a lo sumo un campo. Los algoritmos de la NTIA y el CPqD realizan un control sobre posibles offsets temporales, pero el escenario de la pérdida de cuadros es más complejo y no fue tenido en cuenta.

Es importante tener en cuenta que el ojo humano percibe una secuencia de cuadros como “continua” a partir de 25 cuadros por segundo. Un humano promedio en general no puede detectar de forma consciente la falta de algunos pocos cuadros en una secuencia.

De lo anterior se desprende que el DMOS resultante de las pruebas subjetivas no se ve mayormente afectado.

Por su parte, todos los algoritmos estudiados hacen una comparación cuadro a cuadro, por lo que no es de extrañar que se vean afectados negativamente por este fenómeno (Ver Anexo G. *Análisis de la influencia de la pérdida de cuadros*).

Esto llevaría a pensar que el DMOS siempre sería menor que el DMOSp, sin embargo la pérdida de cuadros no afecta siempre de igual manera al DMOSp. El perder un cuadro al inicio de la transmisión lleva a que los algoritmos comparen, casi en su totalidad, cuadros desfasados, obteniendo un VQM correspondiente a una degradación muy alta. Sin embargo, si la pérdida se da al final de la transmisión, la cantidad de cuadros desfasados comparados por los algoritmos será mucho menor, afectando menos al VQM. Posteriormente a la obtención del VQM, se realiza el proceso de calibración y ajuste, el cual busca eliminar offset

entre el VQM y el DMOS, obteniéndose el DMOSp. Este proceso puede llevar eventualmente a que los videos con menor degradación puedan incluso obtener un DMOSp menor que el DMOS.

Otros factores que pueden afectar al DMOS y al DMOSp cuando existe una pérdida de cuadros son la cantidad de movimiento del video y el tipo de cuadro perdido (I, B o P).

## Capítulo 6. Bibliografía

1. **ITU.** Recommendation ITU-R BT.1683, Objective perceptual video quality measurement techniques for standard definition digital broadcast television in the presence of a full reference. *Recomendación.* 2004.
2. **Muñoz, Gustavo Sánchez.** *Gusgsm.* [En línea] 2007. <http://www.gusgsm.com>.
3. **Winkler, Stefan.** *Digital Video Quality: Vision Models and Metrics.* s.l. : Wiley, 2005.
4. **Kerr, Douglas A.** *Chrominance Subsampling in Digital Images.* 2005.
5. **ITU.** Recommendation ITU-T T.81, Information technology - Digital compression and coding of continuous-tone still images - Requirements and guidelines. *Recomendación.* 1993.
6. **Joskowicz Zilberfarb, Jose.** *Medida de la calidad perceptual de video.* Montevideo : s.n., 2007.
7. **Wu, H. R. y Rao, K. R.** *Digital Video Image Quality and Perceptual Coding.* London : Taylor & Francis Group, 2006.
8. **Wikipedia.** Motion compensation. *Wikipedia.* [En línea] 17 de Marzo de 2009. [http://en.wikipedia.org/wiki/Motion\\_compensation](http://en.wikipedia.org/wiki/Motion_compensation).
9. **Internet Engineering Task Force, Network Working Group .** RFC1122 - Requirements for Internet Hosts - Communication Layers. 1989.
10. **Internet Engineering Task Force, Network Working Group.** RFC1123 - Requirements for Internet Hosts - Application and Support. 1989.
11. **Internet Engineering Task Force, Network Working Group .** RFC1349 - Type of Service in the Internet Protocol Suite. 1992.
12. **ITU.** Recommendation ITU-T P.910, Subjective video quality assessment methods for multimedia applications. *Recomendación.* Setiembre de 1999.
13. —. Recommendation ITU-R BT.500-11, Methodology for the subjective assessment of the quality of television pictures. *Recomendación.* 2002.
14. **Point Topic.** IPTV Subscribers Q2 2008. *Point Topic.* [En línea] 2008. <http://point-topic.com/myhome.asp>.
15. **Video Quality Experts Group.** *Final report from the Video Quality Experts Group on the validation of objective models of video quality assessment.* s.l. : International Telecommunication Union, 2000.
16. —. *Final report from the Video Quality Experts Group on the validation of Objective models of video quality assessment, Phase II.* s.l. : International Telecommunication Union, 2003.
17. **Pastrana Vidal, Ricardo R. y Gicquel, Jean Charles.** *Automatic Quality Assessment of Video Fluidity Impairments Using a No-Reference Metric.* 2006.
18. **Le Callet, Patrick, y otros.** *No Reference and Reduced Reference Video Quality Metrics for End to End QoS Monitoring.* 2005.
19. **Oelbaum, Tobias y Diepold, Klaus.** *Building a Reduced Reference Video Quality Metric with Very Low Overhead using Multivariate Data Analysis.* s.l. : Institute for Data Processing Technische, Universität München Munich, Germany, 2007.
20. **Wolf, Stephen y Pinson, Margaret H.** *LOW BANDWIDTH REDUCED REFERENCE VIDEO QUALITY MONITORING SYSTEM.* s.l. : Institute for Telecommunication Sciences (ITS), National Telecommunications and Information Administration (NTIA), 2005.
21. **ITU.** ITU-T J.144 Measurement of the quality of service Objective perceptual video quality measurement techniques for digital cable television in the presence of a full reference. s.l. : TELECOMMUNICATION STANDARDIZATION SECTOR OF ITU (03/2004), Marzo de 2004.

22. **Carnec, Mathieu, Le Calleta, Patrick y Barbaa, Dominique.** *Objective quality assessment of color images based on a generic perceptual reduced reference.* Nantes : École Polytechnique de l'Université de Nantes, 2008.
23. **Drelie Gelasca, Elisa y Ebrahimi, Touradj.** *On Evaluating Video Object Segmentation Quality: A Perceptually Driven Objective Metric.* 2006.
24. **Fang, Meng, y otros.** Objective Perceptual Video Quality Measurement using a Foveation-Based Reduced Reference Algorithm. *Multimedia and Expo, 2007 IEEE International Conference on Digital Object Identifier.* 2-5 de Julio de 2007, págs. 308 - 311.
25. **Video Quality Experts Group.** *VQEG.* [En línea] <http://www.its.bldrdoc.gov/vqeg/>.
26. **ITU.** Recommendation ITU-R BT.500-8, Methodology for the subjective assessment of the quality of television pictures. *Recomendación.* September de 1998.
27. —. Recommendation ITU-R BT.500-10, Methodology for the subjective assessment of the quality of television pictures. *Recomendación.* 2000.
28. —. Recommendation ITU-R BT.1676, Methodological framework for specifying accuracy and cross-calibration of video quality metrics. *Recomendación.* 2004.
29. **Wikipedia.** MPEG-2. *Wikipedia.* [En línea] 31 de Marzo de 2009. <http://es.wikipedia.org/wiki/MPEG-2>.
30. —. H.264. *Wikipedia.* [En línea] 18 de Marzo de 2009. <http://es.wikipedia.org/wiki/H.264>.
31. **Joskowicz, José, López Ardao, J.Carlos y González Ortega, Miguel A.** *A Mathematical Model for Evaluating the Perceptual Quality of Video.* s.l. : Future Multimedia Networking 2009, Portugal, 2009.
32. **Wikipedia.** Video. *Wikipedia.* [En línea] 4 de Abril de 2009. [http://es.wikipedia.org/wiki/Señal\\_de\\_vídeo](http://es.wikipedia.org/wiki/Señal_de_vídeo).
33. **ITU.** COM 12 – D 97 – E, Packet Loss Distributions and Packet Loss Models. Ginebra, Suiza : s.n., 27-31 de Enero de 2003.
34. **Video Quality Experts Group.** *FR-TV Full Reference Television Phase II Test Plan, Version 1.1 .* s.l. : ITU, 2001.
35. **Streamcrest.** Streamcrest. [En línea] 2008. <http://www.streamcrest.com/>.
36. **Canopus.** *Canopus.* [En línea] 2009. <http://www.canopus.com>.
37. **VideoLAN team.** *VideoLAN.* [En línea] 2008. <http://www.videolan.org/>.
38. **Shunra Software Ltd.** *Shunra Software.* [En línea] 2008. <http://www.shunra.com/>.
39. **Casas Hernández, Pedro, Guerra Vidal, Diego y Irigaray Bayarres, Ignacio.** *Calidad de Servicio Percibida en Servicios de Voz y Video sobre IP.* Montevideo : s.n., 2005.
40. **OPNET Technologies, Inc.** *OPNET.* [En línea] 2009. <http://www.opnet.com>.
41. **ITU.** Recommendation ITU-R BT.814, Specifications and Alignment Procedures for Setting of Brightness and Contrast of Displays. *Recomendación.* Enero de 1994.
42. —. Recommendation ITU-R BT.815, Specification of a Signal for Measurement of the Contrast Ratio of Displays. *Recomendación.* Enero de 1994.
43. **Wallroth, Johannes.** [En línea] 28 de September de 1999. <http://www.programming.de>.
44. **PassMark.** *PassMark Software.* [En línea] 2009. <http://www.passmark.com>.
45. **Nokia.** *Nokia.* [En línea] Setiembre de 2008. <http://www.nokia.com/>.
46. **Sony Creative Software.** *Sony.* [En línea] 2009. <http://www.sonycreativesoftware.com>.
47. **Perera, Gonzalo.** *Probabilidad y Estadística Matemática, un primer encuentro.* Montevideo : Centro de Estudiantes de Ingeniería, 2002.
48. **Horus Grupo Oftalmológico.** *Horus.* [En línea] 2004. <http://www.horusgo.com/>.
49. **Lee, Chulhee.** *Objective Video Quality Assesment.* 2006.

50. **Kuzmic, Dr. Petr.** *BioKin, Ltd.* [En línea] 28 de Agosto de 2007. [www.biokin.com](http://www.biokin.com).
51. **Rodríguez Bocca, Pablo.** *Quality-centric design of Peer-to-Peer systems.* Rennes : Université de Rennes, 2008.
52. **Winkler, Stefan y Campos, Ruth.** Video Quality Evaluation for Internet Streaming Applications. *Human Vision and Electronic Imaging vol. 5007.* 20 de Junio de 2003, págs. 104-115.
53. **Webster, Arthur y VQEG.** *TUTORIAL, Objective perceptual assessment of video quality: Full reference television.* s.l. : ITU, 2004.

## Anexo A. Código fuente del algoritmo avi2yuv

```

function yuv = aviexttoyuv(secuencia,salida,formato)
%AVIEXTTOYUV convierte un video avi(mjpeg)con cuadros negros a
un video YUV formato
%VQEG sin cuadros negros.
%secuencia es el video tal cual es importado por Matlab
%salida es el nombre del archivo YUV que retorna el script
%formato es un vector que indica si el video es de 625 lineas o
525 lineas,
%los valores posibles son '625' y '525'
%-----Inicializacion-----
empezo_negro=0;
empezo_video=0;
numframe=1;
tamaño=size(secuencia(1).cdata);
m=tamaño(1); %numero de filas
n=tamaño(2); %numero de columnas
c=strcat(salida, '.yuv');
yuv=fopen(c, 'w');
%-----
%-----Busco primer cuadro negro-----
while empezo_negro==0;
    frame=secuencia(numframe).cdata;
    a=(frame(:, :, :)<10);
    b=sum(sum(a));
    if b > (m*n)*.90
        empezo_negro=1;
    else numframe=numframe+1;
    end
end
%-----
%-----Busco primer cuadro con contenido-----
while empezo_video==0;
    frame=secuencia(numframe).cdata;
    a=(frame(:, :, :)<10);
    b=sum(sum(a));
    if b > (m*n)*.90
        numframe=numframe+1;
    else empezo_video=1;
    end
end
inicio=numframe+10;
%-----
%-----Determino el largo del video -----
if strcmp(formato, '625')
    fin=inicio+199;
elseif strcmp(formato, '525')
    fin=inicio+239;
else
    error('Formato no reconocido')
end
%-----
%-----Armo el archivo YUV-----

```

```
for k=inicio:fin
    image422=zeros (2*n*m,1);
    image444=rgb2ycbcr (secuencia(k).cdata);
    for i=0:m-1
        for j=0:2:n-1
%submuestreo y reordenado
            image422 ((i*n +j)*2 +1) = image444(i+1,j+1,2);
            image422 ((i*n +j)*2 +2) = image444(i+1,j+1,1);
            image422 ((i*n +j)*2 +3) = image444(i+1,j+1,3);
            image422 ((i*n +j)*2 +4) = image444(i+1,j+2,1);
        end
    end
    fwrite(yuv,image422,'uint8');
end
fclose(3)
%-----
end
```

## Anexo B. Composición de las secuencias de video

A continuación se detalla la conformación de cada una de las cuatro secuencias proyectadas en las pruebas subjetivas.

Secuencia 625-1						
Posición		Video	Cantidad de movimiento	Tasa de codificación (Mbtis/s)	Probabilidad de pérdida de paquetes (%)	Posición del video dañado
1	4	Moving graphic	Baja	2	1	B
2	9	Rugby	Alta	3	5	A
3	7	Fries	Media	2	1	B
4	2	Barcelona	Media	1.2	2	B
5	3	Harp	Media	1.5	5	A
6	4	Moving graphic	Baja	2	1	B
7	5	Canoa Valsesia	Alta	3	0.5	A
8	6	F1 Car	Alta	2	1	B
9	7	Fries	Media	3	2	A
10	8	Horizontal scrolling 2	Alta	0.9	5	B
11	9	Rugby	Alta	2	2	A
12	10	Mobile&calendar	Media	3	0.5	B
13	2	Barcelona	Media	2	0.5	A
14	3	Harp	Media	3	1	B
15	4	Moving graphic	Baja	0.9	2	A
16	5	Canoa Valsesia	Alta	1.2	1	B
17	7	Fries	Media	1.2	5	A
18	8	Horizontal scrolling 2	Alta	1.5	1	B
19	9	Rugby	Alta	3	5	A
20	10	Mobile&calendar	Media	0.9	0.5	B
21	6	F1 Car	Alta	Sin codificar	0	NC
22	2	Barcelona	Media	3	1	A
23	3	Harp	Media	0.9	1	B
24	4	Moving graphic	Baja	1.2	5	A
25	5	Canoa Valsesia	Alta	1.5	2	B
26	6	F1 Car	Alta	1.5	0.5	A

Tabla B 1. Composición de la secuencia de videos 625-1

En esta secuencia de videos se puede observar que se insertó el video 6 "F1 Car" sin degradar, de forma de evaluar la percepción de los observadores a un par de videos sin degradar, de Alta velocidad, en formato NTSC.

Secuencia 625-2						
Posición	Video		Cantidad de movimiento	Tasa de codificación (Mbtis/s)	Probabilidad de pérdida de paquetes (%)	Posición del video dagradado
1	5	Canoa Valsesia	Alta	2	5	B
2	7	Fries	Media	0.9	2	A
3	4	Moving graphic	Baja	2	1	B
4	2	Barcelona	Media	0.9	1	B
5	3	Harp	Media	1.2	2	A
6	4	Moving graphic	Baja	1.5	0.5	B
7	5	Canoa Valsesia	Alta	2	5	A
8	6	F1 Car	Alta	3	2	B
9	7	Fries	Media	0.9	2	A
10	8	Horizontal scrolling 2	Alta	1.2	0.5	B
11	9	Rugby	Alta	1.5	1	A
12	10	Mobile&calendar	Media	2	5	B
13	2	Barcelona	Media	1.5	5	A
14	3	Harp	Media	2	0.5	B
15	4	Moving graphic	Baja	3	2	A
16	5	Canoa Valsesia	Alta	0.9	0.5	B
17	6	F1 Car	Alta	1.5	0.5	A
18	7	Fries	Media	2	1	B
19	8	Horizontal scrolling 2	Alta	3	5	A
20	9	Rugby	Alta	1.2	0.5	B
21	4	Moving graphic	Baja	Sin codificar	0	NC
22	10	Mobile&calendar	Media	1.5	2	A
23	7	Fries	Media	1.5	0.5	B
24	10	Mobile&calendar	Media	Sin codificar	0	NC
25	8	Horizontal scrolling 2	Alta	2	2	A
26	10	Mobile&calendar	Media	1.2	1	B

Tabla B 2. Composición de la secuencia de videos 625-2

En esta secuencia se puede observar que se insertaron los videos 4 “Moving graphic” y 10 “Mobile&calendar” sin degradar, para evaluar la percepción de los observadores ante un par de videos sin degradación con cantidad de movimiento Baja y Media respectivamente en formato NTSC.

También se puede observar que el video colocado en la posición 17 de la secuencia 625-2, se repite en la posición 26 de la secuencia 625-1. Este video se insertó en las dos secuencias de forma de contar con un video testigo para el formato NTSC (625 líneas) evaluado por todos los observadores.

Secuencia 525-1						
Posición	Video		Cantidad de movimiento	Tasa de codificación (Mbtis/s)	Probabilidsad de pérdida de pauqetes (%)	Posición del video dagradado
1	16	Betes_pas_betes	Baja	1.5	2	A
2	22	Tempete	Media	1.2	2	B
3	17	Le_point	Alta	1.2	5	A
4	13	Baloon-pops	Alta	1.2	1	B
5	14	NewYork 2	Baja	1.5	5	A
6	15	Mobile&Calendar	Media	2	2	B
7	16	Betes_pas_betes	Baja	3	0.5	A
8	17	Le_point	Alta	0.9	2	B
9	18	Autumn_leaves	Baja	1.2	2	A
10	19	Football	Alta	1.5	5	B
11	20	Sailboat	Baja	2	2	A
12	21	Susie	Baja	3	2	B
13	22	Tempete	Media	0.9	1	A
14	13	Baloon-pops	Alta	2	5	B
15	14	NewYork 2	Baja	3	1	A
16	15	Mobile&Calendar	Media	0.9	5	B
17	16	Betes_pas_betes	Baja	1.2	1	A
18	17	Le_point	Alta	1.5	0.5	B
19	18	Autumn_leaves	Baja	2	0.5	A
20	19	Football	Alta	3	1	B
21	13	Baloon-pops	Alta	Sin codificar	0	NC
22	20	Sailboat	Baja	0.9	5	A
23	21	Susie	Baja	1.2	5	B
24	15	Mobile&Calendar	Media	Sin codificar	0	NC
25	22	Tempete	Media	1.5	5	A
26	14	NewYork 2	Baja	1.2	2	B
27	16	Betes_pas_betes	Baja	2	5	A
28	18	Autumn_leaves	Baja	0.9	1	B
29	20	Sailboat	Baja	1.5	1	A
30	22	Tempete	Media	3	1	B

Tabla B 3. Composición de la secuencia de videos 525-1

En esta secuencia se puede observar que se insertaron los videos 13 “Baloon-pops” y 15 “Mobile&Calendar” sin degradar, para evaluar la percepción de los observadores ante un par de videos sin degradación con cantidad de movimiento Alta y Media respectivamente, en formato PAL.

Secuencia 525-2						
Posición	Video		Cantidad de movimiento	Tasa de codificación (Mbtis/s)	Probabilidad de pérdida de paquetes (%)	Posición del video dañado
1	13	Baloon-pops	Alta	1.2	1	A
2	15	Mobile&Calendar	Media	0.9	5	B
3	20	Sailboat	Baja	2	2	A
4	13	Baloon-pops	Alta	1.5	2	B
5	14	NewYork 2	Baja	2	0.5	A
6	15	Mobile&Calendar	Media	3	5	B
7	16	Betes_pas_betes	Baja	0.9	0.5	A
8	17	Le_point	Alta	1.2	5	B
9	18	Autumn_leaves	Baja	1.5	5	A
10	19	Football	Alta	2	0.5	B
11	20	Sailboat	Baja	3	5	A
12	21	Susie	Baja	0.9	2	B
13	22	Tempete	Media	1.2	2	A
14	13	Baloon-pops	Alta	3	0.5	B
15	14	NewYork 2	Baja	0.9	1	A
16	15	Mobile&Calendar	Media	1.2	0.5	B
17	16	Betes_pas_betes	Baja	1.5	2	A
18	17	Le_point	Alta	2	1	B
19	18	Autumn_leaves	Baja	3	1	A
20	19	Football	Alta	0.9	1	B
21	13	Baloon-pops	Alta	2	5	B
22	20	Sailboat	Baja	1.2	0.5	A
23	21	Susie	Baja	1.5	0.5	B
24	20	Sailboat	Baja	Sin codificar	0	NC
25	22	Tempete	Media	2	0.5	A
26	15	Mobile&Calendar	Media	1.5	1	B
27	17	Le_point	Alta	3	2	A
28	19	Football	Alta	1.2	2	B
29	21	Susie	Baja	2	1	A
30	22	Tempete	Media	0.9	0.5	B

Tabla B 4. Composición de la secuencia de videos 525-2

En esta secuencia de videos se puede observar que se insertó el video 20 "Sailboat" sin degradar, de forma de evaluar la percepción de los observadores a un par de videos sin degradar, de Baja velocidad, en formato PAL.

También se puede observar que el video colocado en la posición 21 de la secuencia 525-2, se repite en la posición 14 de la secuencia 525-1. Este video se insertó en las dos secuencias de forma de contar con un video testigo para el formato PAL (525 líneas) evaluado por todos los observadores.

## Anexo C. Código fuente del algoritmo de Yonsei

```

%% Implementación del algoritmo de Yonsei. Edge Peak Signal to Noise
Ratio.
%
% Esta función implementa el algoritmo recomendado en el anexo B
% de la norma ITU-T J.144 y en el anexo 3 de la norma R-REC-BT.1683.
% Ambas resultado del estudio de la VQEG Fase II.
% Algoritmo propuesto por la Universidad de Yonsei, Republica de
Corea.
%
% Tomando como entradas dos videos en todos los formatos soportados
por
% Matlab mediante la función mmreader. Se entiende que se evaluará un
video
% procesado o degradado (Video2), contra una referencia original
(Video1).
% La salida de la función será VQM (Video Quality Metric), Metrica de
0
% (indistinguible con el original, Video1) a 1 (completamente
diferentes)
% y el nombre del video procesado.
%
% La funcion se basa en el calculo del PSNR unicamente sobre los
bordes del
% video original.
%
% Ejemplo:
% function [SEC,VQM] =
Yonsei('test_original.avi','test_procesado.avi',Th)
%
% SEC = Nombre de la secuencia procesada a comparar.
% VQM = Metrica de comparación según el algoritmo de la universidad de
Yonsei.
% Th = "Threshold", umbral de detección de bordes.
%
%

function [SEC,VQM] = Yonsei(Video1,Video2,Th)

SEC=Video2;
VQM=0;

%%Cargo el video original-----
---
VideoOrig = mmreader(Video1);
numFramesOrig = VideoOrig.NumberOfFrames;

%%
%%Cargo el video degradado-----
----
VideoDeg = mmreader(Video2);
numFramesDeg = VideoDeg.NumberOfFrames;

%%
if (numFramesOrig == numFramesDeg)
    for i=1:numFramesOrig

        %%Genero la mascara de bordes con la imagen original.

```

```

%Extraigo un cuadro del video original en RGB.
CuadroOrig = im2uint16(read(VideoOrig, [i]));

%Cambio de RGB a YCbCr.
CuadroYCbCrOrig = rgb2ycbcr(CuadroOrig);

%Genero una imagen con solo la luminancia.
BaseOrigRGB =
ones(VideoOrig.Height,VideoOrig.Width,3,'uint16');
BaseOrigRGB(:,:,1) = CuadroYCbCrOrig(:,:,1);
BaseOrigRGB(:,:,2) = CuadroYCbCrOrig(:,:,1);
BaseOrigRGB(:,:,3) = CuadroYCbCrOrig(:,:,1);
OrigY = rgb2gray(BaseOrigRGB);

%Genero la mascara con los bordes del frame del video original.
HT = fspecial('sobel');
HV=HT';

GradOrig = im2uint16(imfilter(imfilter(OrigY,HT),HV));

ImaskOrig = GradOrig > Th;

%
% subplot(2,2,1);
% imshow(GradOrig);title('Gradiente de gradiente');
% subplot(2,2,2);
% imshow(GradM);title('Suma de Gadientes');
% subplot(2,2,3);
% imshow(ImaskOrig);title('Mascara de Gradiente sobre
gradiente');
% subplot(2,2,4);
% imshow(ImaskM);title('Mascara de Suma.');
```

%%

```

%%Trabajo con el video degradado.
%Extraigo un cuadro del video degradado en RGB.
CuadroDeg= im2uint16(read(VideoDeg, [i]));

%Cambio de RGB a YCbCr.
CuadroYCbCrDeg = rgb2ycbcr(CuadroDeg);

%Genero una imagen con solo la luminancia.
BaseDegRGB = ones(VideoDeg.Height,VideoDeg.Width,3,'uint16');
BaseDegRGB(:,:,1) = CuadroYCbCrDeg(:,:,1);
BaseDegRGB(:,:,2) = CuadroYCbCrDeg(:,:,1);
BaseDegRGB(:,:,3) = CuadroYCbCrDeg(:,:,1);
DegY = rgb2gray(BaseDegRGB);

GradDeg = im2uint16(imfilter(imfilter(DegY,HT),HV));

ImaskDeg = GradDeg > Th;

%
% Muestro el cuadro extraido en colores RGB.
% subplot(2,5,1);
% imshow(CuadroOrig);title('Original, RGB');
% Muestro la imagen en escala de grises (Solo la Y).
```

```

% subplot(2,5,2);
% imshow(BaseOrigRGB);title('Original, Luminancia');
% %El original otra vez.
% subplot(2,5,3);
% imshow(OrigY);title('Original Y');
% %Muestro el cuadro degradado.
% subplot(2,5,4);
% imshow(DegY);title('Degradado Y');
% %Muestro los gradientes obtenidos.
% subplot(2,5,5);
% imshow(GradOrig);title('Gradiente Original');
% subplot(2,5,6);
% imshow(ImaskOrig);title('Mascara Original');
% subplot(2,5,7);
% imshow(GradDeg);title('Gradiente Degradado');
% %Muestro la mascara.
% subplot(2,5,8);
% imshow(ImaskDeg);title('Mascara degradado');
% %El degradado otra vez.
% subplot(2,5,9);
% imshow(BaseDegRGB);title('Degradado RGB');
% %Muestro la Diferencia enmascarada.
% subplot(2,5,10);
% imshow(ImaskOrig.*(double(OrigY-DegY)));title('Diferencia
enmascarada');

```

```

%Edges Pixels en comun entre el video degradado y el video
original.

```

```

EPcomi(i)=sum (sum(ImaskOrig.*ImaskDeg));
%Edges pixels del original y del degradado.
EPSRCi(i) = sum(sum(ImaskOrig));
EPHRCi(i) = sum(sum(ImaskDeg));

SEe(i) = (sum(sum(abs(ImaskOrig).*((double(OrigY-DegY)).^2))));

```

```

end

```

```

% %Análisis de todo el video.

```

```

n = length(EPSRCi);
if n~=numFramesOrig
    'Error con la cantidad de cuadros.'
else
    EPSRC = sum(EPSRCi);
    EPHRC = sum(EPHRCi);
    EPcom = sum(EPcomi);
    MSEe = sum(SEe)/EPSRC;
    Ppico=2^16;
    EPSNR=10*log10(Ppico^2/MSEe);

%POST AJUSTES.
if (EPSNR <= 0) %& (EPSNR >= 60)
    'error tipo 2, EPSNR fuera de rango';
elseif (EPSNR >= 40)
    MEPSNR = EPSNR * 0.8;
elseif (EPSNR >= 35)
    MEPSNR = EPSNR * 0.9;

```

```
elseif (EPSNR >= 25)
    MEPSNR = EPSNR;
elseif ((EPcom/EPsrc)^2 <= 0.35) & ((EPHRC/EPsrc) <= 0.13)
    MEPSNR = EPSNR - 60*(0.1225 - (EPcom/EPsrc)^2) ;
else
    MEPSNR = EPSNR;
end;

%Escalado final.
VQM = 1- 0.02*MEPSNR;

end;

else
    'Error, diferente cantidad de cuadros en video original y
degradado';
end;
toc
```

## Anexo D. Código fuente para el cálculo de las Métricas

El función de MATLAB (*modelo.m*) fue utilizado para realizar el ajuste de los VQM de cada modelo al DMOSp.

```
function dmosp=modelo(coef,vqr)
dmosp=coef(1)/(1+exp(-coef(2).*(vqr-coef(3))));
end
```

El siguiente código de Matlab (*analisis.m*) calcula las 7 métricas expuestas en el informe del VQEG Fase II. Estas métricas fueron utilizadas para evaluar el comportamiento de los modelos propuestos en ese mismo estudio. Parte de este código fue tomado, y modificado, directamente de la recomendación ITU-R BT.1676 *Methodological Framework for Specifying Accuracy and Cross-Calibration of Video Quality Metrics* [28].

```
function resultado=analisis(punt_sub,VQM,signo)
%ANALISIS realiza el analisis estadistico propuesto por el VQEG de los
%valores obtenidos por un algoritmo medida de calidad de video
perceptual.
%   RESULTADO=ANALISIS(punt_sub,VQM,signo)
%   Los parametros de entrada son:
%
%       punt_sub:   Matriz de observaciones subjetivas normalizada.
%                   En el sentido horizontal cambia el observador y en
el
%                   vertical cambia el video observado.
%                   Las diferentes columnas son los diferentes
observadores
%                   y las diferentes filas los diferentes videos.
%                   Si para algun video no hay observacion,
%                   el valor debe ser -1
%
%       VQM:       Vector de resultados arrojados por el algoritmo
%
%       signo:     Signo de la derivada del VQM respecto al DMOS,
debe
%                   d(VQM)/d(DMOS). Si la pendiente es positiva, signo
%                   ser 1 y -1 en caso contrario. Por ejemplo, si el
VQM es
%                   el PSNR, signo debe ser -1.
%
%   RESULTADO es un vector que contiene el valor de cada uno de los 7
test
%   propuestos por la VQEG fase II
%%
% El primer paso es calcular la cantidad de observaciones en cada
video
% buscando todos los valores distintos de -1
close all
obsv=sum(punt_sub~-1,2);
VQM=VQM*signo;
% Luego calculo la Media y la Desviacion Estandar sin tener en cuenta
los
% valores "-1"
DMOS=zeros(size(punt_sub,1),1);
VARIANZA=zeros(size(punt_sub,1),1);
```

```

for i=1:size(punt_sub,1)
    for j=1:size(punt_sub,2)
        if (punt_sub(i,j)~= -1)
            DMOS(i)=DMOS(i)+punt_sub(i,j)/obsv(i);
        end
    end
end
for i=1:size(punt_sub,1)
    for j=1:size(punt_sub,2)
        if (punt_sub(i,j)~= -1)
            VARIANZA(i,1)=VARIANZA(i,1)+((punt_sub(i,j)-
DMOS(i))^2)/(obsv(i)-1);
        end
    end
end
end
DESVEST=sqrt(VARIANZA);
%Antes de calcular cada una de las metricas, es necesario ajustar los
%resultados de VQM a una funcion no lineal de la forma
% dmosp=a/(1+exp(b*(vqm-c)))
%llamada "modelo.m" para obtener DMOSp, valores a ser comparados con
DMOS
coef=nlfit(VQM,DMOS,@modelo,[1 1 1]);
DMOSp=modelo(coef,VQM);

%A continuación se grafican los valores experimentales de DMOS y VQM y
el ajuste realizado
figure(1)
x=[min(VQM):.001:max(VQM)];
y=modelo(coef,x);
plot(signo*VQM,DMOS,'.r',signo*x,y,'-b')
xlabel('VQM')
ylabel('DMOS')
legend('Valores experimentales', 'Funcion ajustada')
grid on
%Tambien se grafican los valores del DMOSp y el DMOS, resaltando los
puntos que son outliers.
%outliers
DMOSp_out=[];
DMOS_out=[];
DESV_out=[];
%no outliers
DMOSp_nout=[];
DMOS_nout=[];
DESV_nout=[];

for i=1:size(punt_sub,1)
    if
        ((DMOSp(i)>DMOS(i)+2*DESVEST(i)/sqrt(obsv(i))) || (DMOSp(i)<DMOS(i)-
2*DESVEST(i)/sqrt(obsv(i))))
            DMOSp_out=[DMOSp_out DMOSp(i)];
            DMOS_out=[DMOS_out DMOS(i)];
            DESV_out=[DESV_out 2*DESVEST(i)/sqrt(obsv(i))];
        else
            DMOSp_nout=[DMOSp_nout DMOSp(i)];
            DMOS_nout=[DMOS_nout DMOS(i)];
            DESV_nout=[DESV_nout 2*DESVEST(i)/sqrt(obsv(i))];
        end
    end
end
figure(2)
plot([0 1],[0 1],'g')
hold on

```

```

%Se modifica el aspecto de las barras de error
E_out=errorbar(DMOSp_out,DMOS_out,DESV_out, '.r');
hold on
E_nout=errorbar(DMOSp_nout,DMOS_nout,DESV_nout, '.b');
hold on
plot(DMOSp,DMOS, '.k', 'markersize', 12)
Eout_c= get(E_out, 'Children');
Enout_c= get(E_nout, 'Children');
ebarXData_out=get(Eout_c(2), 'XData');
ebarXData_nout=get(Enout_c(2), 'XData');
ebarXData_out(4:9:end) = ebarXData_out(1:9:end) - 0.005;
ebarXData_out(7:9:end) = ebarXData_out(1:9:end) - 0.005;
ebarXData_out(5:9:end) = ebarXData_out(1:9:end) + 0.005;
ebarXData_out(8:9:end) = ebarXData_out(1:9:end) + 0.005;
set(Eout_c(2), 'XData', ebarXData_out);
ebarXData_nout(4:9:end) = ebarXData_nout(1:9:end) - 0.005;
ebarXData_nout(7:9:end) = ebarXData_nout(1:9:end) - 0.005;
ebarXData_nout(5:9:end) = ebarXData_nout(1:9:end) + 0.005;
ebarXData_nout(8:9:end) = ebarXData_nout(1:9:end) + 0.005;
set(Enout_c(2), 'XData', ebarXData_nout);
axis ([0 max([max(DMOSp),1]) 0 1]);
xlabel ('DMOSp')
ylabel ('DMOS')
axis equal
axis square
grid on
hold off

%El primer test es el test de correlacion de Pearson
TestPearson=corr(DMOS,DMOSp);
%El segundo test es el test de Spearman
TestSpearman=corr(DMOS,DMOSp, 'type', 'spearman');
%El tercer test es el promedio de outliers, valores que estan a mas de
%2 sigma del DMOS
Outliers=((DMOSp>DMOS+2*DESVEST./sqrt(observ)) | (DMOSp<DMOS-
2*DESVEST./sqrt(observ)));
TestOutliers=mean(Outliers);
%%
%Parte del codigo que sigue a continuación, que implementa las
metricas %4, 5 y 6 fue obtenido de la recomendacion "ITU-R BT.1676
Methodological %Framework for Specifying Accuracy and Cross-
Calibration of Video Quality %Metrics"

%Se realizaron modificaciones para evitar el proceso de calibracion,
%normalizacion y ajustes propuesto en la norma y se eliminaron las
lineas %de codigo que no se utilizaron.
%El codigo original se puede obtener en la recomendacion mencionada.

srcxhrc=(1:1:size(punt_sub,1))';
data_in=[srcxhrc srcxhrc DMOSp observ DMOS VARIANZA];
vqm_sign=1;
best=min(DMOSp);
worst=max(DMOSp);
order=2;

%
% Numero de combinaciones src x hrc
num_comb = size(data_in,1);
% Se obtienen los vectores a usar de data_in
vqm = data_in(:,3);

```

```

num_view = data_in(:,4);
mos = data_in(:,5);
variance = data_in(:,6);
% Se utiliza el formato largo para tener mas decimales
format('long');

% Calculo de la metrica 4, Raiz del error cuadratico medio
vqm_rmse_sa = (sum((vqm-mos).^2)/(num_comb-(order+1)))^0.5;
% Calculo de la resolusion del DMOS
vqm_pairs = repmat(vqm,1,num_comb)-repmat(vqm',num_comb,1);
mos_pairs = repmat(mos,1,num_comb)-repmat(mos',num_comb,1);
stand_err_diff = sqrt(repmat(variance./num_view,1,num_comb)+
repmat((variance./num_view)',num_comb,1));
z_pairs = mos_pairs./stand_err_diff;
% Inclusion de todos los valores por encima de la diagonal
delta_vqm = [];
z = [];
for col = 2:num_comb
delta_vqm = [delta_vqm; vqm_pairs(1:col-1,col)];
z = [z; z_pairs(1:col-1,col)];
end
% Correccion del signo de los valores negativos
z_vqm = z;
negs_vqm = find(delta_vqm < 0);
delta_vqm(negs_vqm) = -delta_vqm(negs_vqm);
z_vqm(negs_vqm) = -z_vqm(negs_vqm);

% Se calculan los intervalos del resolving power donde se evaluara el
% nivel de confianza. Se calculan un numero de intervalos (vqm_bins)
entre min(delta_vqm) y %max(delta_vqm).
% Los intervalos se solapan, por lo que se tendran vqm_bins*2-1
intervalos.
cdf_z_vqm = .5+erf(z_vqm/sqrt(2))/2;
vqm_bins = 10; % En cuantos intervalos se calcularan el promedio del
nivel de confianza
vqm_low = min(delta_vqm); % limite inferior de delta_vqm
vqm_high = max(delta_vqm); % limite superior de delta_vqm
vqm_step = (vqm_high-vqm_low)/vqm_bins; % paso del delta_vqm
% posiciones de los limites y centros de los intervalos
low_limits = (vqm_low+vqm_step/2:vqm_high-vqm_step);
high_limits = (vqm_low+vqm_step:vqm_step/2:vqm_high);
centers = (vqm_low+vqm_step/2:vqm_step/2:vqm_high-vqm_step/2);
mean_cdf_z_vqm = zeros(1,2*vqm_bins-1);
for i=1:2*vqm_bins-1
in_bin = find(low_limits(i) <= delta_vqm & delta_vqm <
high_limits(i));
mean_cdf_z_vqm(i) = mean(cdf_z_vqm(in_bin));
end
% VQM resolving power.
resolvingpower=interp1q(mean_cdf_z_vqm',centers',0.75);
% Esta porcion del codigo calcula la metrica 6, errores de
clasificacion.

subj_th = 1.6; % intervalo de confianza de 95%
num_th = 50; % numero de limites de delta_vqm a examinar
vqm_th_list = [vqm_low:(vqm_high-vqm_low)/num_th:vqm_high];
rel_freqs = zeros(vqm_bins+1,4);
for i = 1:num_th+1
vqm_th = vqm_th_list(i);
% Numero de puntos en la region de falso empate

```

```

rel_freqs(i,1) = length(find((delta_vqm < vqm_th) & (subj_th <=
abs(z_vqm))));
% Numero de puntos en la region de falsa diferencia
rel_freqs(i,2) = length(find((vqm_th <= delta_vqm) & (abs(z_vqm) <
subj_th)));
% Numero de puntos en la region de falsa clasificacion.
if (vqm_sign == 1)
rel_freqs(i,3) = length(find((vqm_th <= delta_vqm) & (z_vqm <= -
subj_th)));
else
rel_freqs(i,3) = length(find((vqm_th <= delta_vqm) & (z_vqm >=
subj_th)));
end
end
% Normalizacion del total para obtener numeros entre 0 y 1
rel_freqs = rel_freqs/length(z_vqm);
% Calculo de la frecuencia de aciertos
rel_freqs(:,4) = (1-sum(rel_freqs(:,1:3)'))';
%Busco el maximo de los valores correctos por encima del DeltaVQM
[fila,col]=find(vqm_th_list>resolvingpower,1);
maximo=0;
for i=col:size(vqm_th_list,2)
    if maximo<rel_freqs(i,4)
        maximo = rel_freqs(i,4);
    end
end
end

%el siguiente codigo implementa el calculo de la metrica N°7 (Test F)
%propuesta por la VQEG
%
for i=1:size(punt_sub,1)
    for j=1:size(punt_sub,2)
        if (punt_sub(i,j)~= -1)
            err_opt(i,j)=punt_sub(i,j)-DMOS(i);
            err_DMOSp(i,j)=punt_sub(i,j)-DMOSp(i);
        end
    end
end
metrica7=sum(sum(err_DMOSp.^2))/sum(sum(err_opt.^2));
%Se arma el vector de resultados
resultado=[TestPearson TestSpearman TestOutliers vqm_rmse_sa
resolvingpower (1-maximo) metrica7 ];
end

```

## Anexo E. Tablas de resultados de los test subjetivos

A continuación se observan los resultados obtenidos de las pruebas subjetivas<sup>10</sup>. En el Anexo B. *Composición de las secuencias de video*, en las tablas *Tabla B 1* *Tabla B 2* *Tabla B 3* y *Tabla B 4*, se puede ver cuál de los videos de cada par es el degradado. Es importante recordar que los valores de MOS obtenidos para los tres primero pares proyectados son descartados.

Videos	Versión	Sujetos															
		1	2	3	4	11	12	13	14	15	16	17	18	19	25	26	
4	A	6	7.7	6.4	6.8	6.2	6.3	7.7	7.7	6.4	7.1	8.5	6.7	7.7	7.8	5.2	
	B	4.2	2.3	2.4	4.2	2.3	4.5	5.8	4.5	2.5	3.6	3.7	1.3	4.2	3.1	2.2	
5	A	3.8	2.5	2.3	1.5	0.8	5	1.5	2.6	3.8	2.3	1.5	0.9	3.1	2.1	1.9	
	B	6	7.5	8.8	7.9	6.8	6.5	8.4	8.6	9	7.4	8.3	6.7	7.8	8.5	7.7	
6	A	5	5.2	4.4	7	6.3	2.4	8.4	5.6	6.1	7	7.7	5.6	6.5	4.2	5.7	
	B	3.5	3.7	1.5	3.1	2.3	0.8	5.1	2.3	2	4.5	3.5	3.2	5.6	2.2	3.7	
7	A	6.3	5	3.5	6.9	2.4	7.1	7	2.7	5.7	3.6	6.4	5.7	6.2	3.8	5.3	
	B	6.9	6.9	7	5.2	6.4	8.4	9.5	8.8	8	7	8.4	6.9	8.7	7.3	7.9	
8	A	5.4	9.1	8.5	9.2	6.7	8.3	9.7	7.5	9.5	7.1	9	8.4	7.4	7.7	8.2	
	B	4.2	4.1	2.4	6.4	3.2	6.2	6.9	2.4	5.5	3.8	3.6	3.7	4.4	3.3	2.2	
9	A	4.2	1.1	1	1.6	0.2	6.8	4.9	0.7	1.7	0.8	3.6	3.4	2.1	1.7	1.6	
	B	6.2	7.4	8.6	7.2	6.2	9.1	9.1	9	7.5	7.1	8.4	5.5	5.8	7.7	8.2	
10	A	4.4	8.2	6.3	5.8	5.8	6.8	8.3	9.2	7.5	7.2	7.7	5.3	3.8	6.1	4.2	
	B	1.7	3.8	1.2	1.1	1.7	4.7	3.5	1.5	4.9	1.8	4.2	0.5	2.2	1	1.9	
11	A	3.1	2.3	1.1	5.4	2.3	4.5	4.3	2.6	4.3	2.2	2.6	3	2.2	1.3	1.8	
	B	4.9	7.4	6.5	7.4	4.4	5.8	8.4	9	7.5	7.4	8.5	7.7	5.4	6.1	8.3	
12	A	6.5	9.2	6.5	4.7	7.3	8.8	9.5	8.2	7.5	7.1	9.2	5.2	8.2	8.5	5.8	
	B	4.3	4.7	2.4	2.8	4.2	6.3	6.4	4.4	4.3	4.7	3	3.7	5.4	4.1	2.8	
13	A	5.6	6.1	2.4	3.3	4.4	5.7	5.7	4.3	5	4.3	5.4	5.1	4.5	5.3	4.2	
	B	7.7	8.1	6.4	5.2	4.8	8.5	8.5	7.7	7.1	6.7	5.3	5.8	6.5	6.3	6.3	
14	A	6.8	7.9	9.5	8.2	8.5	6.9	8.5	8.9	7.6	7	7.7	7.4	8.6	8.4	6.6	
	B	4.1	4.6	2.4	4.5	2.9	5.5	4.4	3.4	5	3	3.8	4.3	5.3	3.7	3.1	
15	A	3.8	3.5	1.1	1.5	2.2	0.5	6.3	2.4	2.4	4.2	3.3	2.3	2.9	2.3	3.3	
	B	5.2	7.6	6.3	6.5	6.2	1.7	9.7	6.4	7.7	7	7.7	5	6.2	5.8	5.1	
16	A	7.6	7.9	8.9	7.6	6.8	7.7	9.5	9.5	9.7	7.1	9.1	6.9	7.6	8.3	7.2	
	B	4.4	5.4	1.4	4.3	3.5	5.5	2.4	2.8	4.3	4.2	5.7	2.3	4.6	2.3	3.9	
17	A	3.6	0.5	1.2	1.4	0.2	4.5	0.6	1.5	1.9	1.7	1.7	0.9	3.4	2.7	1.6	
	B	6.7	7.8	7.1	7.6	7.5	7.8	9.8	8.8	6.6	7	8.4	7.4	7.6	8.8	8.3	
18	A	4.2	8.1	4.6	4.6	7.7	6.5	9.5	7.7	7.1	7.1	8.4	5.3	6.2	5.6	3.8	
	B	2.9	5.2	2.3	1.5	2.3	4.7	5	2.6	3	4.3	5.6	2.5	4.3	3.7	1.9	
19	A	3.4	0.2	0.5	0.9	0.2	3.3	2.3	0.4	2.3	1.2	0.6	0.3	1.7	0.2	0.9	
	B	4.5	7.8	7.1	7.5	4.4	4.7	9.6	8.5	9.5	7.1	7.3	7.3	6.5	8.3	7.8	
20	A	6.5	9.7	7.1	6.4	7.4	8.3	7.8	8.9	6.5	6.9	9.8	5.2	8.3	6.1	6.7	
	B	5.4	6.1	1.5	1.5	5.5	5.1	3	4.8	2.5	4.4	4.2	6.3	6.1	3.6	4.8	
21	A	7.7	8.5	9.1	8.7	5.7	7.7	8.8	8.7	9.2	6.8	8.2	8.4	7.5	8.3	8.2	
	B	7.7	9.3	9.2	8.4	5.6	7.7	8.8	8.5	9.2	6.8	8.2	8.4	6.4	8.3	8.2	
22	A	5.2	3.7	1.3	4.2	0.7	4.3	5	2.5	4.2	3.4	4.2	4.3	5.8	4.5	4.2	
	B	8	9.6	6.9	5.7	6.3	6.3	9.5	7.5	7.4	6.6	8.4	6.9	8.2	8.3	7.7	
23	A	6.4	8.1	7.1	8.4	7.3	6.7	9.5	8.7	8	6.8	7.9	7.1	8.7	6.9	7.8	
	B	5.6	4.1	2.6	3.7	2.4	2.7	3.7	2.8	3.7	3.3	4.2	3.1	5.7	3.4	3.8	
24	A	2.5	2.5	0.5	1.6	0.3	0.5	2.3	1.5	2	3.2	3.7	0.7	1.5	0.8	1.7	
	B	4.9	8.4	6.8	5.8	6.2	1.7	6.7	6.6	7.7	6.7	6.8	6.5	6.6	6.2	5.2	
25	A	6.9	8.1	6.7	7.7	5.7	8.3	9.7	8.2	9.7	6.9	8.4	6.6	7.4	8.2	7.1	
	B	4.8	3.8	0.7	4.3	1.8	6.3	2.2	1.7	1.7	3	4.3	2.3	1.8	2.2	0.8	
26	A	2.8	3.8	1.3	2.2	0.3	5	0.4	1.2	0.6	1.5	3.6	0.5	3.1	3.4	0.5	
	B	6.9	7	8.3	8.4	4.2	7.3	9.7	7.4	9.7	6.7	8.2	7.7	6.7	8.3	8.2	

Tabla E 1. Resultados obtenidos de las pruebas subjetivas para la secuencia 625-1

<sup>10</sup> Recordar que los valores obtenidos del sujeto número 7 fueron descartados según los criterios expuestos en 3.6.2

Videos	Version	Sujetos													
		5	6	7	8	9	10	20	21	22	23	24	27	28	29
4	A	6.3	4.3	6.3	7.4	5.2	7.6	6.2	7.8	5.8	8.3	9.6	8.9	7.7	6
	B	3.5	1.7	5.1	3.7	3.1	6.5	1.8	3.6	3.1	2.7	3.4	4.4	2.2	3.7
5	A	3.4	0.6	6.5	4.4	3	4.6	6.2	2.6	2.3	3.7	3.1	3.2	3.2	4.2
	B	4.9	6.3	7	8.4	6.5	7.8	7.7	7.7	8.2	8.8	9.8	8.6	9.2	8.6
6	A	5.2	2.4	6.3	6.8	6.1	7	6.7	6.9	6.1	7.7	9.2	5.4	8.5	6.5
	B	3.2	1.8	5.8	4.5	3.9	3.1	3.8	5.2	1.8	4.8	3.6	4.6	1.6	2.7
7	A	2.2	0.5	3.8	3.5	2.3	1.7	0.3	1.9	1.5	3.8	1.6	3.5	1.1	2
	B	7	6.2	6.6	8.3	7.3	6.3	8.2	8.2	7.8	8.1	9.7	7.7	8.7	7.7
8	A	4.9	4.8	6.7	8.6	7.6	8.2	8.2	8.4	7.8	9.4	9.7	8.3	9.5	7.7
	B	3.3	1.7	4.5	3.6	3.3	3.4	1.8	5.6	1.8	4.4	1.7	2.4	1.7	2.7
9	A	3.1	0.2	3.6	2.4	3.1	4.2	0.8	1.6	2.2	3.7	3.4	4.7	2.5	2
	B	5.5	7.8	4.7	7.3	6.7	7.7	7.6	7.9	7.7	7.8	9.7	7.3	9	7
10	A	6.3	4.2	5.1	7.4	6.5	6.2	6.9	7.6	1.8	7	9.5	8.2	8.5	5.2
	B	4.4	1.8	5	4.3	4.1	4.2	4.5	6.2	0.2	3.8	3.7	3.7	2.6	3.8
11	A	4.2	1	5.8	4.3	3.5	4.2	4.5	8.2	1.2	2.5	1.6	1.7	1.4	2.3
	B	7.5	6.3	5.8	7.6	7.2	7.8	7	7.3	7.8	6.9	9.6	7.5	7.7	8.2
12	A	6.7	7.5	4.7	8.3	6.8	7.1	7.8	7.4	7	8.5	9.8	9.2	8.5	6.5
	B	2.9	0.9	5	3.2	2.9	3	1.2	4.5	1.7	1.9	1.6	2.3	2.2	2.2
13	A	3.7	0.9	6.1	3.5	3.2	2.3	1.8	3.4	1.8	1.9	2.3	0.9	2.4	2
	B	6.7	7	6.5	7.7	5.7	7.6	6.2	6.9	6.2	6.5	9.8	8.5	7	6.4
14	A	6.3	7.3	6.5	8.4	7.5	7	7.2	7.3	8.2	8.2	9.7	7	9.2	5.6
	B	3.6	0.9	6.3	5	3.8	5.5	4.2	4.8	2.9	3.8	4.4	2.3	2.4	3.1
15	A	1.5	0.3	3.4	2.5	2.1	2.3	0.8	3.8	1.9	1.8	1.5	1	1.8	2.6
	B	6.5	3.8	4.2	7	5.3	6.3	6.5	7.5	7	7.6	9.6	7.2	8.2	5.7
16	A	5.7	7	7.3	7.3	7.2	7.7	8.1	7.8	6.2	8.2	9.5	8.4	9.6	8.2
	B	3.1	0.5	5.8	4.4	1.7	2.5	1.4	5.4	1.8	3	4.2	4.8	2.3	3.9
17	A	3.3	0.3	4.8	4.6	1.9	3.5	0.5	3	1.9	4.8	1.7	2.8	6.5	2.5
	B	5.5	7.6	6.7	8.3	7.3	7.6	7.7	7.7	7.6	8.8	9.7	9.5	9.5	8.4
18	A	2.4	7.4	6.8	8.5	7.3	6.7	8.2	8.4	6.5	7.6	9.6	8.9	9.5	7.7
	B	2.9	2.6	6.2	4.5	3.5	4.8	5.1	5	2.3	3.8	3.7	4.5	6.3	3.2
19	A	3	0.1	4.2	1.6	1.7	1.7	0.7	1.3	0.3	1.9	1.6	2.4	0.5	2.3
	B	6.4	4.7	5.7	6.5	7	6.2	6.9	7.8	2.2	7.8	9.3	5.1	9.4	4.3
20	A	6.7	5.8	6.7	8.3	7.3	6.9	6.1	7.4	7.5	8.5	9.6	8.9	9.1	6.9
	B	3.2	0.8	6.2	4.3	2	1.8	2.1	3.7	2.2	2.5	3.8	2.4	3.8	3.5
21	A	4.8	5	6.2	7	6.9	7.6	7.2	7	7.8	7.7	8.9	8.2	8.5	7.2
	B	4.5	4.6	6.3	7	6.7	6.3	6.4	7	7.8	7.7	7.4	7.5	9.4	7.2
22	A	5.7	1.5	5.7	3.4	3.3	3.1	2.1	3.4	1.8	1.9	2.5	3.1	2.4	2
	B	4.3	5.5	6.1	8.4	6.3	6.9	7.1	7.2	7.7	7.8	9.5	7.5	7.7	6.8
23	A	6.4	6.3	6.6	8.5	7.4	7.7	8.2	7.6	7.8	7.8	9.6	9.1	9.2	4
	B	4.4	3.7	6.4	5	6.1	4.9	6.1	4.9	3.8	6.9	5.6	4.5	4.2	4.4
24	A	6.3	5.5	6.2	8.4	7.5	7.3	7.8	7.4	7.7	8.6	9.5	8.9	8.3	4.6
	B	6.2	5.2	6.2	8.3	7.5	6.6	7.8	7.1	7.7	7.5	9.3	8.9	8.3	6
25	A	3.2	0.4	5.3	2.6	2.4	3.6	0.4	1.8	1	4.4	1.7	2.6	1.8	1.7
	B	5.7	4.4	5.1	6.8	7.2	6.3	4.9	7.4	2.2	7.8	9.5	7.6	8.3	4.7
26	A	6.6	5.3	6.1	8.3	6.4	7.1	7.4	7.2	7.4	7.8	9.5	8.2	7.5	5.3
	B	4.4	0.4	5.5	2.6	5.1	4.3	2.4	3	2.7	3.8	3.8	4.8	2.3	3.4

Tabla E 2. Resultados obtenidos de las pruebas subjetivas para la secuencia 625-2

Videos	Versión	Sujetos																									
		1	2	3	4	11	12	13	14	15	16	17	18	19	25	26											
4	A	7.5	5	5.3	6.7	8.4	3	7	6.6	2.6	7.6	7.9	5.8	6.7	3.6	5											
	B	4.9	1.1	0.5	0.9	1.4	1.2	5	2.3	2	5.7	4.3	1.8	4.6	0.3	0.6											
5	A	4.3	1	0.8	0.5	1.7	4.5	6.4	3	4.4	2.7	1.9	3.8	4.4	3.6	1.8											
	B	9.3	9.3	8.3	6.5	8.3	5.6	9.4	7.1	8	7.7	8.4	5.8	7.9	9.1	8.7											
6	A	7.6	5	6.5	3.4	8.4	5.5	7.6	6.3	6.2	6.9	7.3	5.8	6.3	3.8	6.3											
	B	6	2.5	1.4	2.6	6.7	4.7	5.8	2.7	3.2	4.1	3.7	3.8	4.5	1.3	2.4											
7	A	6.1	3	2.5	4.5	2.6	5.7	6.8	2.8	6	3.7	3.2	5.2	6.1	6.1	4.6											
	B	7.6	7.2	5.7	7.5	8.4	7.4	8.6	7.5	6.8	6.3	7.7	7.3	8.3	7.8	6.2											
8	A	7.2	5.1	5	4.9	9.3	7.1	9	6.5	6.1	8.5	7.8	7.8	5.6	6.1	7.2											
	B	5	1.2	0.7	0.8	3.5	3.7	5	3.5	4	1.7	4.2	2.3	1.7	2.7	0.3											
9	A	4	5.2	0.5	5.8	1	5.4	6.4	2.9	5.4	3.7	2.2	3.6	4.1	0.7	3.4											
	B	8.3	7.5	6.7	7.2	9.4	6.6	9	7	9.1	7.6	8.3	5.8	6.5	5.8	8.3											
10	A	6.2	8.8	7.6	8.2	5.6	8.5	10	9.5	9.5	7.5	8.5	9.5	8.6	8.3	8.3											
	B	3.8	1	0.5	2.8	1.8	6.4	2.8	2.4	5.4	1.3	1.8	0.8	4.3	0.2	1.4											
11	A	6.5	5.2	1.6	4.7	1.3	5.7	5.7	1.7	4.7	4.7	5.4	5.6	6.2	3	2.1											
	B	8.4	6.8	7.6	7	7.1	6.9	7.7	5.5	7.7	8.9	8.3	7	7.7	7	7.3											
12	A	7.6	7.3	8.5	9.2	8.4	9.4	9.8	9.2	8.3	8.2	8.6	8.4	8.7	8.5	7.4											
	B	4.2	3.3	1.3	5.5	3.1	7.4	3.5	4.7	5.8	3.5	3.8	2.7	5.7	1.2	0.8											
13	A	7.2	5.2	3.2	5.1	2.3	1.6	5.7	7	4.2	5.8	5.9	7	5	3.3	5.2											
	B	8.5	8.8	5.5	7.8	7.4	1.7	7.5	8.4	7	7.5	8.1	7.1	6.2	7.5	7.7											
14	A	7.5	4.6	5.5	6.8	6.3	2.4	6.9	6.8	6.9	8.2	8.6	4.9	6.4	3.8	4.2											
	B	3.5	1.3	0.2	1	2.4	1.4	1.7	1.2	2	2.3	1.8	3.1	1.7	0.2	1.7											
15	A	4.4	5.1	1	5.6	2.3	5.7	5	3.3	5.8	4.4	6.2	5	6.2	1.4	4.9											
	B	8.8	8.5	7	7.5	6.3	6.8	7.2	7.7	8.2	8.2	9.2	7.2	7.8	6.7	9.3											
16	A	7.7	7.1	7	4.5	6.7	2.5	6.2	6.4	5.6	6.2	8.4	5.2	6.3	6.1	5.1											
	B	4.4	1	0.5	3	3	1.7	1	0.7	1.7	2.1	4.3	0.7	4	0.2	1.8											
17	A	5.3	5.1	1.4	6.4	2.1	3.2	5.7	2.5	4.1	2.4	3.2	4.7	4.7	3.1	3.6											
	B	7.8	7.4	7.3	8.2	4.2	3.8	8.3	8.4	7.8	7	7.8	7.5	6.5	7.2	5.3											
18	A	7.6	6.7	7.5	3.5	7.4	6.3	8.6	8.3	7.8	7.7	9.1	7.7	7.8	8.5	6.3											
	B	6.9	5	2.8	1.5	2.3	4.7	5.9	3.6	4.2	3.6	7.6	4.4	6.1	2.3	2.5											
19	A	6.3	2.8	2.8	5.3	2.9	3.6	5.9	2.6	4.9	4.2	6.2	4.8	6.3	3.3	5.3											
	B	8.5	9	7.1	7.6	8.3	5	9	7.7	7.7	7.5	9	7.8	4.4	8.2	7.8											
20	A	7.7	9	8.9	8.2	3.6	9.3	9.9	8.4	10	7.6	8.6	9.3	4.7	9.4	9.6											
	B	6.3	2.5	5.1	7.2	1.1	8.2	7.7	4.2	6.5	3.8	6.3	5	6.7	4.2	5.9											
21	A	7	6.8	7	7.7	6.4	1.8	9.1	6.3	7.5	7.3	8.9	5	5.7	4.3	4.7											
	B	6.9	4.9	7	7.7	5.5	1.8	7.7	5.3	6	7.3	8.3	5.1	6.4	5.7	5.8											
22	A	5.6	1	1.2	5.6	2.3	2.7	5.5	3.5	5	3	4.8	3.5	5.7	3.1	3.8											
	B	7.5	7	7.2	7.3	7.5	3.4	8.7	7.6	7.8	6.8	8.4	7.7	7.8	8.4	7.4											
23	A	6.9	8.7	8.8	8.5	8.8	9.5	7.6	9.3	8.4	7.4	9.2	8.4	8.4	9.2	7.6											
	B	4.2	0.7	0.5	3.3	1.4	7.8	0.6	1.6	4.6	1.8	3.4	3.5	4.3	1.6	1.7											
24	A	7	6.7	7.4	6.2	8.3	1.5	7.6	7.6	6.8	6.4	7.7	4.7	6.7	4.8	6.3											
	B	6.7	8.5	7.4	6.2	8.3	1.8	7.6	8.3	6.8	6.4	7.7	4.7	6.4	5.8	6.3											
25	A	5.7	1.1	0.5	3.3	1.2	1.5	2.8	0.9	2	1.4	4.4	2.7	3.8	2.2	2.4											
	B	7.6	8.6	4.3	6.9	5.6	2.7	6.5	6.6	6	7.2	8.4	4.9	5	7.7	7											
26	A	8.5	9	6.5	7.5	8.5	8.6	8.9	7.7	7.8	7.9	8.8	7	7.5	5.7	8.9											
	B	4.4	0.9	0.6	5.7	2.5	6.4	5.7	1.4	3.6	2.3	3.7	3.7	5	3.8	3.8											
27	A	3.7	1	0.6	1.7	2.2	0.8	1	0.7	4.2	7.8	1.1	0.8	1.3	0.5	1.6											
	B	6.2	7.3	6.6	7.7	4.5	1.6	6.9	8.4	7.8	7.8	7.5	8.4	5.6	7.4	7.1											
28	A	8.3	7.4	7.7	6.2	6.9	5.8	8.7	8.3	8	7.5	8.4	7.1	6.6	8.2	7.8											
	B	4.6	4.8	4.3	4.6	2.7	4.6	5.7	3.3	5.8	4.1	3.7	5.2	5.7	4.7	4.9											
29	A	5.7	2.8	4.3	5.7	0.7	4.4	6.4	1.4	6	4.1	2.3	3.3	5.2	4.9	3.8											
	B	7.6	9.1	8.4	7.7	6.5	5.5	8.4	8.7	8	7.3	7.8	5.2	7.7	8.2	6.5											
30	A	7.5	7.1	6.6	5.5	6.5	1	8.4	7.6	7.6	7.2	7.8	6.8	7.9	8.2	7.3											
	B	6	2.9	2.5	3.7	3.2	1.4	6.7	3.2	3.8	4.5	5.6	5.6	5.6	4.7	5.5											

Tabla E 3. Resultados obtenidos de las pruebas subjetivas para la secuencia 525-1

Videos	Versión	Sujetos															
		5	6	7	8	9	10	20	21	22	23	24	27	28	29		
4	A	7.2	4.8	6.3	4.4	6.2	7.6	7	5.2	7.6	7.9	9.4	8.3	8.6	7.4		
	B	2.9	0.6	2.9	2.4	2.4	3	3.7	1.7	0.7	3	3.3	1.1	2.7	1.2		
5	A	2.8	0.8	6.2	3.7	5	3.5	4.4	2.5	4.2	3.8	4.3	3.8	3	3		
	B	6.7	6.3	7.2	5.5	7.4	7.8	5.8	6.9	7.8	8.3	9.8	7.7	8.9	6.3		
6	A	6.9	4.4	6.7	4.9	5.5	7	6.5	6.8	1.2	4.9	5	7.6	5.2	6		
	B	1.2	0.8	3.6	2.4	2	2.3	1.7	2.3	8.2	2.3	0.9	3.5	0.9	1.9		
7	A	6.4	3.2	6.5	5.6	6	5.6	0.5	6.7	5	7.6	4.8	5.4	6.9	5.6		
	B	6.6	6.3	6.5	5.7	7.1	7.2	2.7	7.5	7	8	7.7	6.7	9	6.4		
8	A	6.9	6.5	6.3	6.3	8.4	7.7	7.1	6.5	7	7.2	9.1	7.6	6.8	6		
	B	2.8	0.6	3.7	2.5	1.7	2.2	1.5	1.7	4.2	2.7	1.3	3.7	0.8	2.3		
9	A	3	0.8	3.7	2.3	4.9	3.7	2.4	1.8	4.2	3	3.1	3.1	2.8	2.8		
	B	6.8	6.8	6.2	5.5	6.5	7.8	6.4	7.8	7.9	8	9.6	8.4	9	8		
10	A	7	6.9	6.6	7.2	6.4	8.2	7.8	7.2	8.2	8.3	9.6	8.4	8.9	8.6		
	B	2.7	0.7	4.2	4.7	3.6	4.9	4.8	4.5	5.9	3.9	3.1	6	2.8	3.7		
11	A	3	0.9	4.4	2.6	2.5	2.2	3.2	1.4	1.7	2.5	2.4	0.7	1.2	1.8		
	B	6.9	6.8	6.4	5.4	5.4	7.1	5.5	6.7	6.9	5.7	9.7	6.9	7	8		
12	A	8.9	7	6.5	7.1	8.7	7.5	6.4	7	7.9	8.2	9.7	9.2	9	9		
	B	2.9	1.1	4.6	4.4	4.2	4.2	4.7	2.9	5	4.4	2.5	3.5	3.1	3.1		
13	A	2.8	1.1	6.2	4.9	4.3	4.9	1.7	4.5	4.8	2.5	3.5	2.4	3	3.5		
	B	6.5	7.5	6.5	6.3	6.3	6.5	7.8	7.4	7.9	7.5	9.4	7.8	7	6		
14	A	7	6.5	4.2	5.7	4.9	7.5	5.7	7.1	5.8	6.5	6.8	5	4.9	5.7		
	B	2.6	1	3.8	3.6	3	5.6	4.2	3.5	2.2	2.6	2.4	4.3	2.5	2.5		
15	A	2.9	1.3	6.7	4.4	6.1	6.7	6.3	5	7.5	3.8	3.7	3.4	4.9	4.3		
	B	6.7	7.3	7.2	6.6	7	7.8	7.7	7.5	8.3	7.8	9.1	7	8.7	5.7		
16	A	7.2	5.4	6.1	6.7	4.3	7.4	7	7.4	7	5	7.7	6.8	8.8	4.3		
	B	2.8	0.7	5.8	4.7	3.1	5.5	2.2	2	4.2	3.6	2.7	5.2	4.8	2.6		
17	A	2.8	0.7	5.2	3.4	2.4	4.4	6.2	3	1.8	3.8	2.4	1.6	1	2		
	B	7.4	5.7	6.3	5.7	5.5	7.1	7.8	7.8	7.8	7.5	8.4	8.8	8.7	5.7		
18	A	7	5.7	6.6	6.6	7.1	6.4	7.3	7.1	4.3	7	9.2	6.5	8.8	6		
	B	5	1.7	6.5	6.7	6.7	5.4	6.2	6.6	8.5	7.1	5.7	4.3	5	5.5		
19	A	3	0.6	4.7	3.7	4.4	4.2	4.8	4.5	1.8	3.8	4.3	4.5	2.9	3.8		
	B	6.8	5.7	6.1	5.7	5.8	7.7	7.7	7.8	6.2	8.2	9.5	7.7	9	4.7		
20	A	6.4	7.7	6.6	7.1	8.2	7.6	8.3	7.9	9.1	8.2	9.4	9.1	8.7	8.1		
	B	0.9	0.4	3.8	1.7	1.7	2.4	0.2	1.1	0.9	2.8	1.6	3.7	1.2	2		
21	A	7	6.7	4.2	7	4.8	7	6.3	7.7	5.8	8.2	9.3	8.5	7	6.4		
	B	1.5	0.6	2.3	1.5	1.8	1.8	2.7	2.9	0.9	1.5	1.5	0.4	1.2	1.6		
22	A	6.5	1.7	6.2	3.6	2.8	5.2	5	5.7	5.8	3.8	4.6	4.5	4.8	6		
	B	6.9	5.7	6.3	5	5	7.6	5.7	7.1	7	6.8	9.6	7	8.8	7.5		
23	A	7	7.9	6.6	7	8.2	7.6	7.2	7.8	7.8	8.3	9.5	8.8	8.9	7.7		
	B	4.6	5.3	6.4	5.8	4.2	4.4	6.7	5.9	6.2	6.4	5	6.7	6.7	3.6		
24	A	6.8	5.4	6.3	5.6	4.9	6.8	6.3	8.2	6.2	7	9.5	5.5	4.8	8.7		
	B	7.2	5.5	6.3	5.6	4.7	7.5	6.6	7.8	6.2	7.7	7.1	5.5	7	8		
25	A	7	2.8	6.1	6.6	3.5	5.5	5.7	6.5	6.1	7.8	9.5	7.5	4.8	4.4		
	B	6.6	6.4	6.2	6.2	7.5	6.9	6.3	7.8	7.2	7.8	6.3	7.4	9	3.4		
26	A	6.8	4.4	6.1	6.2	4.6	6.4	7.7	7.7	5.5	7.2	9.5	5.7	6.8	6		
	B	4.8	0.5	5.8	3.6	3.5	4.4	2.2	3.5	3.9	3.2	3.3	4.5	2.8	3.3		
27	A	2.8	1	4.3	3.5	2.1	4.9	0.3	3.7	2.9	3.8	3.2	3.4	0.9	2		
	B	3.1	6.3	6.1	6.9	7	7.7	8.2	7.6	7.8	7.8	9.3	7.7	9	7.4		
28	A	7	7.7	6.4	7	7.7	8.2	8.9	8.3	7	7.7	9.5	9.7	9	7		
	B	3.2	0.4	3.8	2.6	2	3	0.9	5.3	1.5	3.9	2.8	2.7	2.9	2.6		
29	A	3.2	1.2	4.4	4.3	4.7	4.4	6.2	5.4	4.1	4.5	5	5.7	4.9	4		
	B	4.8	7.1	6.3	7	7.7	8.2	7.7	8.1	7	7.8	9.4	8.4	9.4	7.2		
30	A	7.2	7.3	6.1	5.5	7.2	6.3	5.5	7.7	7	7	9.4	5.4	8.5	6		
	B	3.2	1.4	5.7	4.4	5.7	3.5	6.5	5.1	3.8	3.8	5.3	5.5	3.6	4.5		

Tabla E 4. Resultados obtenidos de las pruebas subjetivas para la secuencia 525-2

## Anexo F. Tablas de resultados de los test objetivos

En las tablas se pueden observar los resultados devueltos por cada uno de los algoritmos para cada de videos. En las tablas se puede ver cada uno de los parámetros del sistema al que fueron sometidas las secuencias para su degradación.

Estos datos fueron sometidos luego al tratamiento indicado en 3.6.4.

Secuencia	Formato	Cantidad de movimiento	Tasa de compresión (CBR)	% de pérdida de paquetes	VQR CPqD	VQR NTIA	Yonsej	PSNR
src13_ref_525	525	Alta	3000	0.5	0.4507	0.4166	0.5108	21.6663
src13_ref_525	525	Alta	1200	1	0.6834	0.6592	0.5537	19.3949
src13_ref_525	525	Alta	1500	2	0.7551	0.7675	0.5909	17.3176
src13_ref_525	525	Alta	2000	5	0.8029	1.0054	0.6396	15.0648
src14_ref_525	525	Baja	900	1	0.2789	0.3013	0.3403	33.3067
src14_ref_525	525	Baja	3000	1	0.5442	0.2756	0.4809	22.9009
src14_ref_525	525	Baja	2000	0.5	0.5532	0.2832	0.4786	23.0028
src14_ref_525	525	Baja	1200	2	0.7130	0.5297	0.5297	20.6142
src14_ref_525	525	Baja	1500	5	0.7673	0.7336	0.5918	17.8817
src15_ref_525	525	Media	2000	2	0.6818	0.8538	0.6526	14.7921
src15_ref_525	525	Media	1200	0.5	0.7287	0.4950	0.5431	20.4230
src15_ref_525	525	Media	1500	1	0.5086	0.4365	0.5725	18.8584
src15_ref_525	525	Media	900	5	0.8533	1.0201	0.7127	12.6877
src15_ref_525	525	Media	3000	5	0.8268	1.0541	0.6753	13.7239
src16_ref_525	525	Baja	900	0.5	0.2194	0.4401	0.3621	29.9342
src16_ref_525	525	Baja	3000	0.5	0.0403	0.1463	0.3201	36.4229
src16_ref_525	525	Baja	1200	1	0.1708	0.3976	0.4306	26.7747
src16_ref_525	525	Baja	1500	2	0.2746	0.3931	0.5172	22.2828
src16_ref_525	525	Baja	2000	5	0.4437	0.7702	0.5701	19.7846
src17_ref_525	525	Alta	2000	1	0.3408	0.4059	0.4031	27.3481
src17_ref_525	525	Alta	1500	0.5	0.4024	0.4825	0.4174	27.3516
src17_ref_525	525	Alta	900	2	0.8015	0.8145	0.6606	15.0952
src17_ref_525	525	Alta	3000	2	0.7379	0.6815	0.6279	16.6979
src17_ref_525	525	Alta	1200	5	0.8257	1.0644	0.6793	14.2126
src18_ref_525	525	Baja	900	1	0.3130	0.4008	0.3355	30.5014
src18_ref_525	525	Baja	3000	1	0.3956	0.2419	0.3530	29.6639
src18_ref_525	525	Baja	2000	0.5	0.3932	0.3415	0.4196	27.0405
src18_ref_525	525	Baja	1200	2	0.5612	0.5815	0.4663	24.0503
src18_ref_525	525	Baja	1500	5	0.5132	0.5671	0.4622	24.5596
src19_ref_525	525	Alta	3000	1	0.1642	0.2258	0.3281	31.3273
src19_ref_525	525	Alta	2000	0.5	0.3213	0.3078	0.3736	28.0441
src19_ref_525	525	Alta	1200	2	0.8049	0.6547	0.5649	18.7167
src19_ref_525	525	Alta	1500	5	0.8610	0.6292	0.6173	16.5869
src19_ref_525	525	Alta	900	1	0.8302	0.6559	0.4965	22.5124
src20_ref_525	525	Baja	1200	0.5	0.1749	0.2162	0.3715	29.4137
src20_ref_525	525	Baja	2000	2	0.1422	0.2841	0.3812	28.7608
src20_ref_525	525	Baja	1500	1	0.1480	0.2154	0.4132	27.9203
src20_ref_525	525	Baja	900	5	0.3339	0.5099	0.6041	19.2014
src20_ref_525	525	Baja	3000	5	0.1859	0.4228	0.6146	18.7569
src21_ref_525	525	Baja	1500	0.5	0.5642	0.2656	0.3304	29.4605
src21_ref_525	525	Baja	2000	1	0.0856	0.1557	0.2980	36.9936
src21_ref_525	525	Baja	900	2	0.7063	0.5069	0.3990	26.0761
src21_ref_525	525	Baja	3000	2	0.6903	0.6297	0.4836	22.5796
src21_ref_525	525	Baja	1200	5	0.7328	0.8852	0.6263	16.7984
src22_ref_525	525	Media	2000	0.5	0.6294	0.5628	0.5385	20.0329
src22_ref_525	525	Media	900	1	0.6099	0.5142	0.4610	24.0527
src22_ref_525	525	Media	900	0.5	0.6043	0.5309	0.4637	24.1611
src22_ref_525	525	Media	3000	1	0.1663	0.1848	0.3429	30.1587
src22_ref_525	525	Media	1200	2	0.6456	0.6003	0.5363	20.1451
src22_ref_525	525	Media	1500	5	0.8155	0.7027	0.5822	17.7987

Tabla F 1. Resultados obtenidos de las pruebas objetivas en formato PAL

Secuencia	Formato	Cantidad de movimiento	Tasa de compresión (CBR)	% de pérdida de paquetes	VQR CPqD	VQR NTIA	Yonsei	PSNR
src2_ref_625	625	Media	2000	0.5	0.4256	0.5985	0.4563	24.5801
src2_ref_625	625	Media	1200	2	0.7835	0.7541	0.5362	20.3509
src2_ref_625	625	Media	3000	1	0.3690	0.3665	0.3867	28.1662
src2_ref_625	625	Media	900	1	0.8652	0.7251	0.5156	21.4023
src2_ref_625	625	Media	1500	5	0.9927	1.0121	0.6004	17.6780
src3_ref_625	625	Media	3000	1	0.8292	0.2735	0.4189	26.3124
src3_ref_625	625	Media	900	1	0.9653	0.6176	0.5449	19.4460
src3_ref_625	625	Media	2000	0.5	0.8850	0.5206	0.5355	19.7185
src3_ref_625	625	Media	1200	2	0.5680	0.4466	0.4019	25.7238
src3_ref_625	625	Media	1500	5	0.9874	0.8658	0.6272	16.1610
src4_ref_625	625	Baja	2000	1	0.2654	0.2487	0.4353	27.6565
src4_ref_625	625	Baja	1500	0.5	0.1837	0.1602	0.3558	33.4523
src4_ref_625	625	Baja	900	2	0.4150	0.3685	0.4709	26.6317
src4_ref_625	625	Baja	1200	5	0.4961	0.8396	0.5712	21.5542
src4_ref_625	625	Baja	3000	2	0.2635	0.4428	0.4935	24.7692
src5_ref_625	625	Alta	3000	0.5	0.2575	0.2740	0.3288	31.1151
src5_ref_625	625	Alta	1200	1	0.6470	0.5070	0.4034	27.3023
src5_ref_625	625	Alta	1500	2	0.9629	0.7181	0.5910	17.6106
src5_ref_625	625	Alta	900	0.5	0.8211	0.6993	0.5372	20.1516
src5_ref_625	625	Alta	2000	5	0.9076	0.7887	0.5964	17.3103
src6_ref_625	625	Alta	2000	1	0.9989	0.5997	0.5568	18.8545
src6_ref_625	625	Alta	3000	2	0.9708	0.4830	0.5351	19.8649
src6_ref_625	625	Alta	1500	0.5	0.9735	0.4924	0.4809	23.1890
src7_ref_625	625	Media	1500	0.5	0.9650	0.4194	0.5074	21.4675
src7_ref_625	625	Media	2000	1	0.8043	0.5625	0.6086	16.4302
src7_ref_625	625	Media	3000	2	0.9880	0.5904	0.5984	16.7803
src7_ref_625	625	Media	900	2	0.9980	0.6979	0.6132	16.3506
src7_ref_625	625	Media	1200	5	0.9998	0.7139	0.6502	14.2713
src8_ref_625	625	Alta	1500	1	0.9142	0.5391	0.5688	18.0613
src8_ref_625	625	Alta	1200	0.5	0.8684	0.5501	0.5593	18.3078
src8_ref_625	625	Alta	900	5	0.9240	1.0375	0.6078	16.5065
src8_ref_625	625	Alta	2000	2	0.9939	1.0328	0.6013	16.6465
src8_ref_625	625	Alta	3000	5	0.9965	1.0391	0.6073	16.5005
src9_ref_625	625	Alta	2000	2	0.9587	0.5658	0.5750	18.0823
src9_ref_625	625	Alta	1500	1	0.9782	0.6423	0.5988	16.8723
src9_ref_625	625	Alta	1200	0.5	0.9998	0.6851	0.6095	16.4140
src9_ref_625	625	Alta	3000	5	0.9762	0.8220	0.6553	14.6532
src10_ref_625	625	Media	900	0.5	0.5924	0.4896	0.4475	25.1819
src10_ref_625	625	Media	3000	0.5	0.2937	0.3195	0.3840	28.1127
src10_ref_625	625	Media	1200	1	0.8251	0.6804	0.6050	17.0547
src10_ref_625	625	Media	1500	2	0.8856	0.8237	0.6211	16.2477
src10_ref_625	625	Media	2000	5	0.9269	1.0232	0.6575	14.6810

Tabla F 2. Resultados obtenidos de las pruebas objetivas para el formato NTSC

## Anexo G. Análisis de la influencia de la pérdida de cuadros

En el presente anexo se estudia brevemente como afecta la pérdida de cuadros al desempeño de los algoritmos estudiados.

Para esto se estudiaron dos secuencias degradadas, en las cuales la pérdida de cuadros es notoria. Las secuencias estudiadas fueron la secuencia 9 (formato PAL) codificada a una tasa de 2.0 Mbps y degradada con 2% de pérdida de paquetes en la transmisión, y la secuencia 13 (formato NTSC) codificada a una tasa de 1.2 Mbps y degradada con un 1% de pérdida de paquetes en la transmisión. Para esta tarea se utilizó el software de *MSU Video Quality Measurement Tool*<sup>11</sup>. Dicho software permite la visualización de forma simultánea de la secuencia original y la degradada, pudiendo entonces observar en qué posición de la secuencia degradada se pierde un cuadro. En la primera secuencia degradada se identificó la falta de un cuadro entre el cuadro #74 y el #75, mientras que en la segunda secuencia entre el cuadro #67 y el #68. Dichas secuencias se modificaron repitiendo el último cuadro anterior a la pérdida (los cuadros #74 y #67 respectivamente).

Con estas secuencias modificadas, los nuevos pares original-procesado fueron sometidos a los algoritmos y el PSNR fue calculado obteniéndose los resultados observados en la Tabla G 1.

Secuencia.	CPqD	Yonsei.	NTIA	PSNR
Secuencia 13	0,7188	0,5537	0,6580	19,3949
Secuencia 13 modificada	0,4216	0,5185	0,6410	21,4915
Secuencia 9	0,8173	0,5750	0,5651	18,0823
Secuencia 9 modificada	0,4038	0,5199	0,5523	21,1822

Tabla G 1. Comparación de resultados obtenidos de los algoritmos

Como era de esperar se observaron diferencias apreciables, siempre tendientes a la estimación de una mejor calidad perceptual. El VQM más afectado por esta corrección fue el obtenido del algoritmo del CPqD, donde se alcanzó una variación de casi 200%. El segundo VQM más afectado fue el obtenido del PSNR, donde la variación fue de un poco más del 10%. El resto de los VQM, si bien se vieron afectados, no presentan variaciones significativas.

Observando la gráfica del PSNR cuadro a cuadro para los pares original/degradado y original/degradado-correcto secuencia 9 (Ver Figura G 1), se puede ver como a partir del cuadro #74 el PSNR de la secuencia corregida mejora notoriamente cuadro a cuadro.

<sup>11</sup> Software libre obtenido de [http://compression.ru/index\\_en.htm](http://compression.ru/index_en.htm)

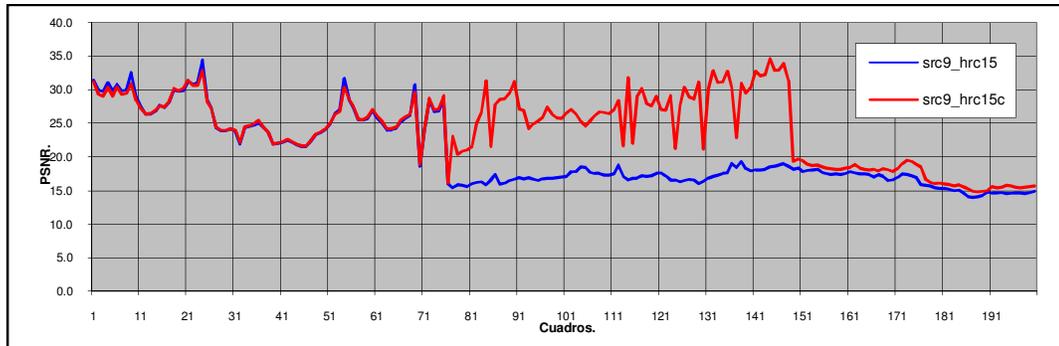


Figura G 1 PSNR cuadro a cuadro para las secuencia 9 y nueve modificada.

A su vez se observa que en el cuadro #147 como el PSNR de la secuencia corregida retoma nuevamente valores similares a los de la secuencia sin corregir. Este último efecto puede ser provocado por una segunda pérdida de cuadro. Otro efecto importante que se ve puede observar es que el efecto generado por la pérdida de un cuadro depende directamente de la posición del cuadro perdido. Si la pérdida del cuadro hubiera ocurrido dentro de las primeras posiciones, la diferencia del PSNR de la secuencia hubiera sido mucho mayor, de la misma manera si el cuadro perdido hubiera sido el último, la diferencia hubiera sido casi nula.

## Anexo H. Contenido del CD

A continuación se detallan los archivos contenidos en el CD que se incluye en la contratapa de esta documentación.

1. Archivo cvprrip.pdf: documentación del proyecto almacenada en formato PDF (Portable Document Format) de Adobe ©.
2. Archivo cvprrip\_paper.pdf: artículo *An Evaluation of the VQEG Algorithms in a Packet Loss Scenario* redactado en idioma inglés y en formato IEEE. Almacenado en formato PDF (Portable Document Format) de Adobe ©.
3. Archivo cvprrip\_poster.pdf: poster ilustrativo del proyecto almacenado en formato PDF (Portable Document Format) de Adobe ©.

NOTA: para poder acceder a los archivos en formato PDF es necesario disponer del programa Adobe Reader ©, versión 6.0 o superior. Al momento de escribir esta documentación se utilizó un computador personal con las siguientes características:

Procesador Intel Centrino 2 de 2.26 GHz con 2.0 GB de memoria RAM.