



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

“OPTIMIZACION DE VIDEO STREAMING SOBRE REDES UMTS”

TESINA DE SEMINARIO

Previo a la obtención del Título de

INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES

Presentado por:

YULL ARTURO MATAMBA VALENCIA

ANDRES XAVIER ROGEL VALAREZO

GUAYAQUIL – ECUADOR

**AÑO
2010**

AGRADECIMIENTO

Agradecemos a nuestros padres y familiares por el apoyo brindado durante toda nuestra vida, ya que sin ellos, este proyecto no se pudo haber llevado a cabo. Han sido pilares fundamentales en la realización de nuestras metas.

También quisiéramos agradecer al Ing. Juan Carlos Avilés y al Ing. Washington Medina, que nos dieron su confianza, su dedicación, sus consejos con los cuales fue posible la culminación de este proyecto.

No nos podemos olvidar de todas las personas que nos dieron su apoyo incondicional: amigos, profesores; que confiaron en nosotros aun en los tiempos más difíciles.

DEDICATORIA

A mis padres que me dieron su apoyo toda mi vida, por ellos me esfuerzo cada día para lograr mis metas; y con esta nueva etapa que llega a mi vida, seguiré honrando la confianza que depositaron en mí.

ANDRES ROGEL VALAREZO

Dedico este proyecto a Dios a mis Padres, a mis hermanas por estar siempre pendiente de mí apoyándome en todo momento y dándome las fuerzas necesarias para continuar adelante

YULL MATAMBA VALENCIA

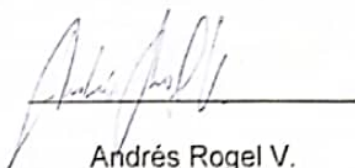
DECLARACION EXPRESA

"La responsabilidad del contenido de este trabajo final de graduación, nos corresponde exclusivamente; y el patrimonio intelectual del mismo a la ESCUELA SUPERIOR POLITECNICA DEL LITORAL"

(Reglamento de exámenes y títulos profesionales de la ESPOL)

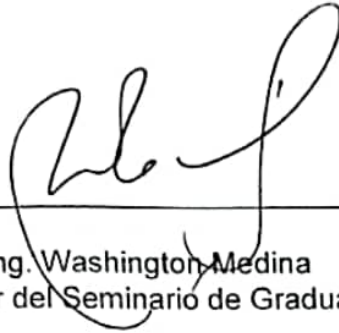


Yull Matamba V.



Andrés Rogel V.

TRIBUNAL DE SUSTENTACION



Ing. Washington Medina
Profesor del Seminario de Graduación



Ing. Juan Carlos Avilés
Delegado del Decano

RESUMEN

El presente estudio aborda el análisis de la optimización de uno de los servicios 3G que ofrece las redes UMTS, el video streaming. El video streaming, es uno de los servicios favoritos que ofrece la revolucionaria tecnología 3G en comparación con su antecesora GSM.

A partir de la manipulación de ciertos parámetros, se producirá la optimización del video para su transporte por la red UMTS de manera confiable y rápida. Esta optimización se analizará con el estudio de ciertas variables que pondrán en manifiesto la diferencia en la calidad del video.

INDICE GENERAL

INTRODUCCION

1 GENERALIDADES UMTS.....	1
2 VIDEO STREAMING SOBRE UMTS.....	4
2.1 CONDICIONES TECNICAS PARA EL VIDEO STREAMING SOBRE UMTS.....	4
2.2 ARQUITECTURA STREAMING.....	6
2.3 INICIALIZACIÓN DE SESIÓN Y CONTROL (H.323, SIP).....	7
2.4 VIDEO STREAMING SOBRE REDES IP.....	8
2.4.1 UDP.....	9
2.4.2 TCP.....	12
2.4.3 RTP.....	15
2.4.4 RTCP.....	20
2.4.5 RTSP.....	22
2.5 JUSTIFICACIÓN DEL FILTRAJE VS ANÁLISIS DE LA FUENTE.....	24
2.6 TIPOS DE SERVICIO.....	27
2.7 PARÁMETROS DE CALIDAD EN EL VIDEO.....	29
3 ESTANDARES DE CODECS PARA VIDEO.....	32
3.1 CODECS UTILIZADOS.....	32
3.2 CODECS PARA AUDIO Y VOZ.....	32
3.3 CODECS PARA VIDEO.....	33
3.4 ANALISIS DE CARACTERISTICAS DESDE TERMINALES ACTUALES.....	35
4 IMPLEMENTACION.....	37
4.1 COMPRESION DE VIDEO UTILIZANDO SIMULINK.....	37

4.2 ADQUISICION DEL ARCHIVO DE VIDEO.....	38
4.3 TRATAMIENTO DE LA SEÑAL DE VIDEO.....	39
4.3.1 BLOQUE DE PROCESAMIENTO.....	39
4.4 SOFTWARE DISPONIBLE PARA TRABAJAR CON VIDEO PARA TERMINALES MOVILES.....	41
4.4.1 PROCEDIMIENTOS PARA LA CODIFICACION.....	43
4.4.2 EJEMPLOS DE CODIFICACION.....	44
5 ANALISIS DE RESULTADOS.....	46
CONCLUSIONES	
RECOMENDACIONES	
REFERENCIAS	
GLOSARIO	

INDICE DE FIGURAS

Figura 1.1 Descripción de la arquitectura de dominio de paquetes.....	3
Figura 1.2 Representación estándar del modelo UMTS.....	3
Figura 2.1 Vista esquemática de una sesión streaming originada vía MMS.....	5
Figura 2.2 Elementos de la red en un servicio streaming de conmutación por paquete 3G.....	7
Figura 2.3 Encapsulamiento de un datagrama UDP dentro de un paquete IP.....	9
Figura 2.4 Un paquete UDP con sus 4 campos de encabezamiento.....	10
Figura 2.5 Impacto de errores en el streaming de video. La imagen muestra una secuencia dispareja (izquierda), con pérdida (en medio), y con reordenamiento de paquetes(derecha).....	11
Figura 2.6 Encapsulación de información TCP en un datagrama IP. El segmento TCP en este diagrama no incluye campos de opciones TCP.....	13
Figura 2.7 Segmento TCP con su campo de encabezado.....	13
Figura 2.8 Encapsulación de un paquete RTP dentro de un datagrama UDP e IP Debe notarse que el encabezado RTP ilustrado no contiene ningún	

CSRC o encabezados de extensión.....	16
Figura 2.9 Un paquete RTP y sus campos de encabezado.....	17
Figura 2.10 Este gráfico muestra como el cliente de medios usa el buffering para manejar la fluctuación (jitter).....	18
Figura 2.11 Un paquete RTCP con sus cinco campos de cabecera.....	21
Figura 2.12 Un ejemplo de una sesión RTSP entre servidor y cliente.....	23
Figura 2.13 Codificador de Imágenes I.....	24
Figura 2.14 Codificador de Imágenes P.....	25
Figura 2.15 Grupo de imágenes.....	26
Figura 2.16 Imagen de referencia n.....	26
Figura 2.17 Imagen n+1.....	26
Figura 2.18 Diferencia entre la imagen n y n+1.....	26
Figura 2.19 Estimación de los vectores de movimiento.....	26

Figura 2.20 Desglose de diferentes niveles en la escala de Calidad de Servicio (QOS).....	28
Figura 2.21 Especificaciones de las necesidades de los diferentes servicios.....	29
Figura 2.22 Esquema del proceso básico para la evaluación mediante DSCQS.....	30
Figura 2.23 Ejemplos de imágenes con diferente grado de ruido.....	31
Figura 3.1 Esquema del proceso de codificación.....	34
Figura 4.1 Esquema de implementación.....	37
Figura 4.2 Video 1.....	38
Figura 4.3 Video 2 antes de la compresión.....	38
Figura 4.4 Bloque “From Multimedia File” (desde el archivo multi medios).....	39
Figura 4.5 Bloque de compresión.....	40
Figura 4.6 Bloque DCT.....	40
Figura 4.7 Video 1 después de la compresión.....	40

Figura 4.8 Video 2 después de la
compresión.....41

Figura 4.9 Software ImTOO para la compresión de
video.....43

INDICE DE TABLAS

Tabla 3.1 Resumen de codecs
soportados.....33

Tabla 3.2 Comparativa de terminales
móviles.....36

Tabla 4.1 Comparativa de software de compresión de
video.....42

Tabla. 4.2 Resultados de la re-codificación de la secuencia de
video.....45

Tabla 5.1 Resultados de la simulación del video
1.....47

Tabla 5.2 Resultados de la simulación del video
2.....47

INDICE DE ECUACIONES

Ecuación 2.1 Expresión para evaluar la PSNR entre
imágenes.....30

INTRODUCCION

La aparición del internet ha permitido desde hace tiempo bajar, reproducir archivos de audio y video de buena calidad. Sin embargo, la transferencia de archivos completos se traduce en tiempos de transferencia muy largos y la imposibilidad de ver y escuchar en tiempo real.

El video streaming es la codificación y el envío de tramas de video entre un servidor y un cliente a través de una red como por ejemplo Internet. Este envío se lo realiza en forma de paquetes fragmentados y encapsulados. En el cliente los paquetes son re ensamblados y posteriormente se procede a reproducir la información multimedia.

El concepto de streaming tiene varias condiciones de manera implícita:

- Se utiliza para transmitir información multimedia en modo de flujos continuos.
- La reproducción en el lado del cliente puede iniciarse desde el momento en que se cuente con datos disponibles, sin necesidad de aguardar hasta que se complete la recepción de todo el flujo.
- No es necesario que el cliente almacene la información que recibe, pudiendo limitarse en la mayoría de los casos a reproducirla.
- En la misma reproducción en el receptor debe mantenerse la misma temporización que la utilizada en la generación del flujo extremo del servidor.

Para optimizar estos servicios de vídeo poseemos dos alternativas básicas de trabajo: optimizar el algoritmo de compresión u optimizar el transporte de la información por la red, siendo la primera alternativa el objeto de estudio del presente trabajo.

En este proyecto, se utilizará una técnica de compresión (Transformada discreta del coseno) para optimizar el transporte de video streaming sobre una red UMTS. El

procedimiento de análisis se centra en la captura de dos videos uno de buena calidad y otro de calidad media a los cuales se realiza la compresión y posteriormente se analizan parámetros como PSNR (Relación Pico Señal a Radio), velocidad de cuadros y radio de compresión, los cuales afectan notablemente la calidad y el transporte de la trama de video por la red. Se ha escogido la herramienta MatLab/SimuLink que nos ayudara en el procesamiento del video y la obtención de los resultados.

Para desarrollar este proyecto, se lo ha dividido en cinco capítulos. En el capítulo 1, es una breve introducción al sistema UMT, su evolución y sus generalidades.

En el capítulo 2, se describe la arquitectura del video streaming sobre UMTS, condiciones técnicas necesarias, protocolos a los que recurre UMTS.

En el capítulo 3, se analiza los codecs (codificadores-decodificadores) utilizados para optimizar el video con un estudio en los terminales actuales.

En el capítulo 4, se realizará una simulación que abarcará la codificación de un video y su respectiva salida después de este proceso.

Y para finalizar en el capítulo 5, se analiza detalladamente los resultados de la simulación.

CAPITULO 1

GENERALIDADES UMTS

Los sistemas de telecomunicaciones móviles se han ido desarrollando evolutivamente definiéndose distintas generaciones. El cambio de una generación a otra lo marcan transiciones sustanciales de la tecnología y las aplicaciones.

Así hemos pasado de la primera generación (1G) de redes móviles analógicas a la segunda generación (2G) de redes con conectividad digital de extremo a extremo como GSM.

En las comunicaciones móviles actuales se demanda mayor capacidad y anchura de banda para poder acceder a nuevos servicios multimedia.

Bajo los auspicios de la UIT se desarrollo la tercera generación (3G) de sistemas móviles, en el que se han definido características de servicios y de calidad de estos

sistemas, recogida en el estándar IMT 2000 (Telecomunicación Móvil Internacional 2000). Uno de los objetivos básicos de IMT 2000 fue el establecer redes que permitieran el acceso a servicios avanzados de cualquier tipología (anykind), desde cualquier lugar (anywhere) y en cualquier momento (anytime). En este marco IMT 2000, se han desarrollado diferentes tipos de multi-acceso a interfaz de radio y de red fija, y se ha optado por la técnica UMTS (Sistema Universal de Telecomunicaciones Móviles), entre cuyos objetivos básicos son los siguientes:

- Sustentación de velocidades de usuario hasta 144 kbit/s, con cobertura y movilidad completas en zonas extensas y hasta 2 Mbit/s, en situaciones de movilidad limitada y cobertura local.
- Presentación de servicios con terminales fijos y móviles, portátiles y de vehículo.
- Elevado rendimiento espectral.
- Alta calidad de voz con tasa de bits reducida.
- Posibilidad de introducción flexible de nuevos servicios, tales como los de datos por paquetes de alta velocidad, multimedia y asimétricos.
- Compatibilidad con los sistemas de tercera generación GSM

UMTS utiliza (W-CDMA). Esta versión de W-CDMA usa propagación directa con un velocidad de chip de 3,84MCPS y un ancho de banda nominal de 5MHz. El modelo soporta uno de los dos modos dúplex W-CDMA: dúplex por división de frecuencia (FDD). Duplex por división de tiempo (TDD) no es compatible.

En el modo FDD, el enlace ascendente y descendente utilizan diferente banda de frecuencia. El radio frame (cuadro de radio) tiene una longitud de 10ms y se divide en 15 ranuras. Los Spreading factors (factor de propagación) varían desde 256 hasta 4 para un enlace ascendente y de 512 a 4 para un enlace descendente.

El dominio de paquetes de la red núcleo incluye dos nodos de red: el SGSN y el GGSN.

El DSN incluye toda la funcionalidad GPRS necesaria para soportar servicios de paquete GSM y UMTS. El SGSN monitorea la ubicación del usuario y realiza funciones de seguridad y control de acceso. El GGSN contiene información de enrutamiento para la conmutación de paquetes (PS) a usuarios conectados y

proporciona interoperabilidad con redes externas PS como la red de paquetes de datos (PDN). El modelo de CN incluye la funcionalidad tanto de los nodos SGSN y GGSN.

EL MSC/VLR se utiliza en la arquitectura de dominio de paquetes para coordinar de manera eficiente los servicios y funcionalidades de PS y CS. El Registro de Ubicación (HLR) contiene información sobre los abonados de GSM y UMTS. La Funcionalidad de Carga de Puerta de enlace (CGF) recoge los registros de carga del SGSN y GGSN. El Registro de Identificador de Equipo (EIR) almacena información sobre la identidad del usuario del equipo.

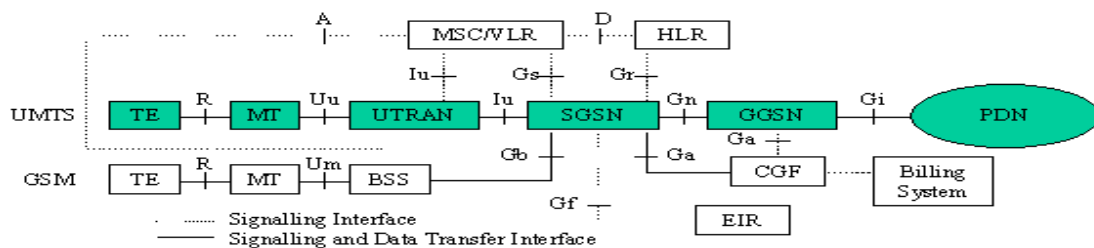


Figura 1.1 Descripción de la arquitectura de dominio de paquetes

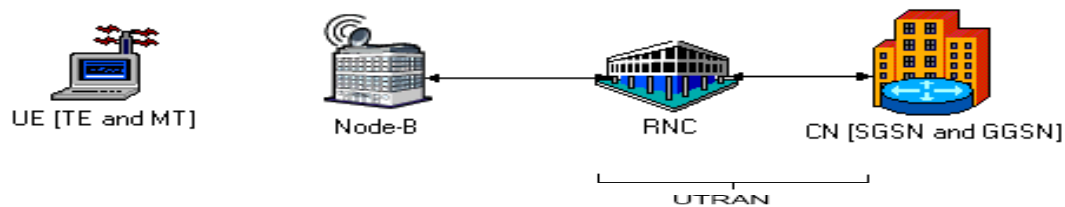


Figura 1.2 Representación estándar del modelo UMTS

CAPITULO 2

VIDEO STREAMING SOBRE UMTS

2.1 CONDICIONES TECNICAS PARA EL VIDEO STREAMING SOBRE UMTS

Streaming es la habilidad que tiene una aplicación de reproducir medios sincronizados como audio y video en una forma continua mientras está siendo transmitida al cliente sobre la red de datos. Para los sistemas de tercera generación (3G), el servicio streaming por paquetes conmutados 3G (PSS) llena el vacío entre los servicios de mensajería multimedia 3G (ej: servicios de conversación y descarga). El video streaming sobre redes UMTS es clasificado como streaming simple. El servicio de streaming simple incluye un conjunto básico de protocolos para control de streaming, protocolos de transporte, codecs y protocolos de descripción. En el caso simple, no hay intercambio de capacidad explícita ni un administrador de encriptación y derechos digitales. Un usuario móvil obtiene un identificador de recursos universal (URI) a un contenido específico que se ajusta a su terminal. Este URI puede venir de un navegador WWW (World Wide WEB), de un navegador WAP (wireless application protocol), o también puede ser tipado.

Este URI especifica un servidor streaming y la dirección del contenido en este servidor. Una aplicación que establece la sesión multimedia debe tomar un archivo SDP (Session Description Protocol). El archivo SDP puede ser obtenido de algunas formas. Puede ser obtenido en un link dentro de la pagina HTML (Lenguaje de Marcado de Hiper Texto) que el usuario descargo. También puede ser obtenido directamente tipado como un URI; también, a través de un protocolo llamado RTSP (real-time streaming protocol). En el caso de una opción de envío streaming en servicios MMS, el archivo SDP es obtenido a través del agente de usuario MMS que recibe un mensaje MMS modificado del servidor MMS.

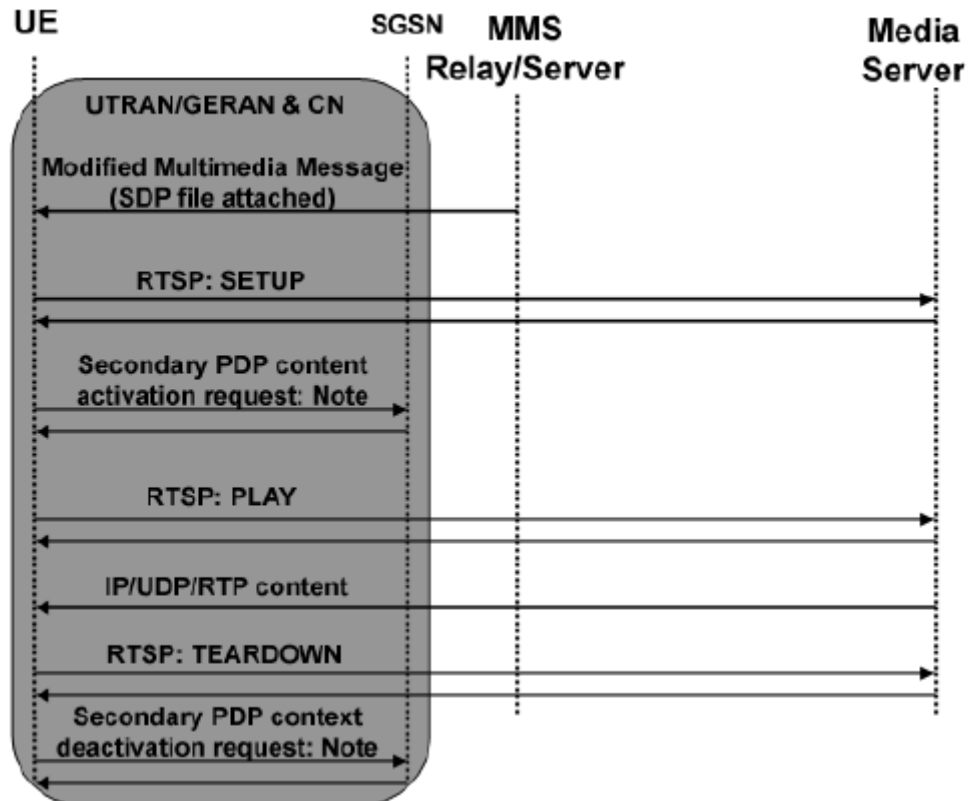


Fig. 2.1 Vista esquemática de una sesión streaming originada vía MMS

El archivo SDP contiene la descripción de la sesión (nombre de sesión, autor, etc.), el tipo de medio a ser presentado, y la tasa de bits. El establecimiento de la sesión es el proceso en el cual el navegador o el usuario móvil “llama” al cliente streaming para que establezca la sesión con un servidor. Se espera que el equipo de usuario tenga un contexto PDP (Packet data protocol) activo en concordancia con el o con otro tipo de portador de radio que habilite la transmisión de paquetes IP cuando recién se establece la sesión. El cliente puede preguntar por más información acerca del contenido. El cliente debe iniciar el aprovisionamiento de un portador con el apropiado QoS (quality of service) para el medio streaming. Se inicia un servicio streaming a través del envío de un mensaje RTSP SETUP por cada medio elegido por el cliente. Esto devuelve el puerto UDP (user datagram protocol) y/o el puerto TCP (transport control protocol) para ser usados por el medio respectivo. El cliente envía un mensaje

RTSP PLAY al servidor que comienza a enviar uno o más streams sobre la red IP. Este caso lo ilustramos en la figura 2.1.

2.2 ARQUITECTURA STREAMING

Un servicio streaming requiere al menos de un servidor de contenido y un cliente streaming. Un servidor streaming está localizado atrás de la interface Gi. Componentes adicionales como portales, servidores de perfil, servidores de almacenamiento y proxy localizados también están detrás de la interface Gi, figura 2.2 pueden estar involucrados para proveer servicios adicionales o para mejorar la calidad de servicio general. Los portales son servidores que permiten accesos al contenido del medio; por ejemplo, un portal puede ofrecer contenido de navegador y facilidades de búsqueda. En el caso más simple, un portal es una página WEB/WAP con una lista de links al contenido streaming. El contenido es usualmente guardado en servidores de contenido, los cuales pueden ser encontrados en cualquier lugar de la red. Los servidores de perfil de usuario y de terminal son usados para guardar preferencias de usuario y capacidades de terminal. Esta información puede ser usada para controlar la presentación del contenido del medio stream para un usuario móvil.

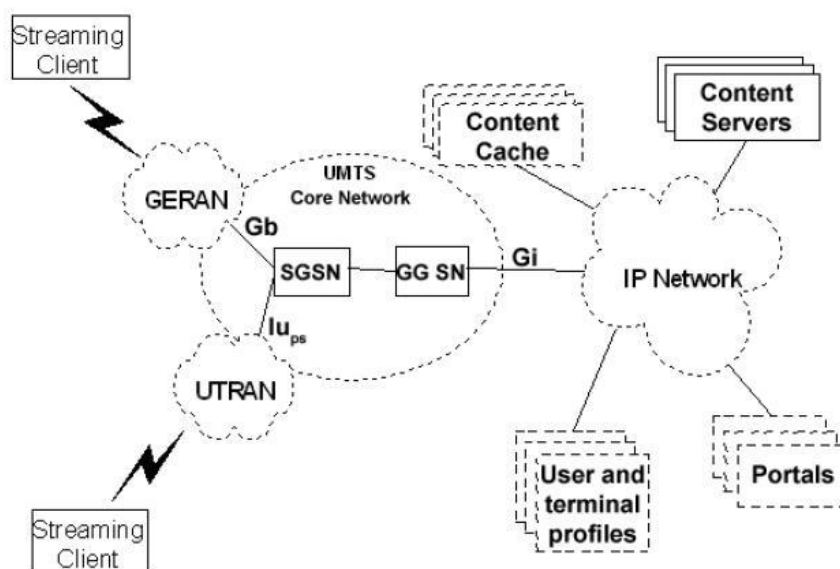


Fig. 2.2 Elementos de la red en un servicio streaming de conmutación por paquete 3G

2.3 INICIALIZACION DE SESION Y CONTROL (H.323, SIP)

El inicio de sesión es requerido cuando se haga streaming en el Internet porque un servidor debe poder identificar los clientes que hacen un pedido y también debe determinar que contenido está siendo requerido.

Adicionalmente, el cliente debe tener ciertas preferencias como, por ejemplo, con que formato hacer el stream, que números de puerto usar para los paquetes, etc. H.323, SIP (Session Initiation Protocol) y RTSP son tres estándares que soportan esta función.

H.323, de la Unión Internacional de Telecomunicaciones (ITU-T), es un estándar que cubre la señalización y el control de las llamadas por Internet. En vez de ser protocolo específico, es más una arquitectura de telefonía en Internet con soporte para contenido multimedia.

SIP fue desarrollado por la IETF (Internet Engineering Task Force), el cual especifica un protocolo que describe como iniciar llamadas telefónicas en Internet, video conferencias y otras conexiones multimedia. A diferencia de H.323, SIP no especifica cómo conducir alguna otra parte del proceso streaming. Aunque hay algunas similitudes entre SIP y H.323 en el cual los dos proveen funcionalidad para habilitar el inicio de sesión, H.323 es más un estándar que especifica como transportar y administrar contenido a más de la configuración de sesión.

RTSP es similar a SIP en que es usado para iniciar una sesión entre un cliente y servidor para permitir comunicación multimedia pero también soporta control

interactivo como grabar y posición absoluta del medio para que un espectador pueda atrasar o adelantar a una escena deseada. Aunque RTSP y SIP coinciden en gran parte, los dos pueden ser usados para que se complementen para una aplicación particular (ej.: aplicación de mensajería).

2.4 VIDEO STREAMING SOBRE REDES IP

UDP o TCP son típicamente usados para el transporte de información sobre redes IP. En esta sección, se examina estos dos protocolos y se identifica problemas asociados a cada uno para el transporte de video streaming. Un protocolo llamado RTP (Real-Time Transport Protocol) que atenta con resolver algunos de estos problemas es discutido más adelante.

2.4.1 UDP

UDP es un protocolo simple de la capa de transporte que es encapsulado en un datagrama IP. Típicamente, cada datagrama UDP es mapeado a un simple paquete IP como podemos observar en la figura 2.3, aunque los datagramas UDP que son más largos que la unidad de tamaño máxima de transmisión (MTU) soportada por la red, puede ser fragmentada en un número de datagramas IP. El tamaño del MTU varía entre las diferentes redes, pero para una red de área local (LAN) es 1500 bytes, siendo la máxima cantidad de información que puede ser llevada en una carga UDP es 1472 bytes y el resto es ocupado por el encabezamiento IP y UDP.

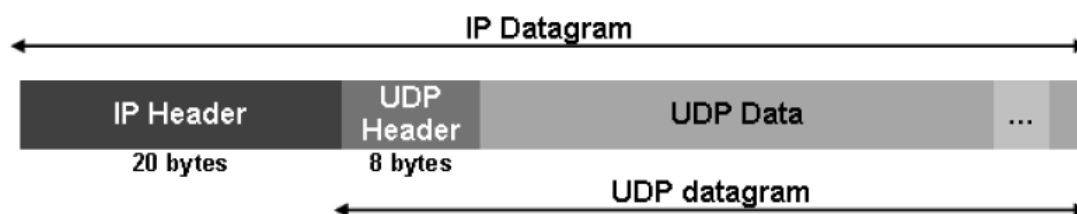


Fig. 2.3 Encapsulamiento de un datagrama UDP dentro de un paquete IP

Como UDP es un protocolo no orientado a conexión, no tiene problemas de overhead (flujo correspondiente a datos de control) asociado con el inicio de conexión. Además,

cada paquete UDP o datagrama consiste de un encabezamiento que es de 8 bytes solamente seguido de la información esto está ilustrado en la figura 2.4. Esto resulta en un overhead de solo 28 bytes por paquete para ambos encabezamientos IP y UDP. Hay cuatro campos de encabezamientos de 16 bits cada uno que en conjunto forman el encabezamiento UDP: el número de puerto de la fuente, el número de puerto del destinatario, el largo del paquete UDP, y el control de suma UDP (control de error).

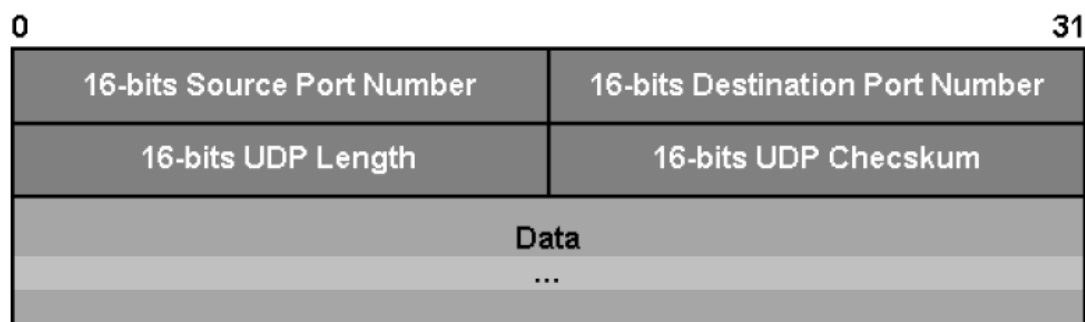


Fig. 2.4 Un paquete UDP con sus 4 campos de encabezamiento

Los números de puerto son usados en este protocolo para que el tráfico pueda ser dirigido a un proceso específico en el cliente. Cada proceso que espera recibir un datagrama UDP debe estar ligado a un número de puerto en particular. El campo del número de puerto de la fuente permite al proceso recibido determinar que puerto puede usar para enviar un paquete de respuesta.

El campo del largo del UDP indica el tamaño de todo el datagrama UDP. La detección de errores en el encabezamiento o la carga (información) puede ser logrado usando el opcional, pero recomendado, campo de suma de control en el encabezamiento UDP. Si la suma de control no es usada, este valor esta en cero.

Como se discutió arriba, UDP es un protocolo muy simple con un bajo overhead de transmisión, pero hay algunos problemas asociados con UDP que son altamente significantes para el streaming de audio y video. Ya que IP es la base para UDP, este

último no es un protocolo confiable que no puede garantizar un buen tiempo de transmisión ni tampoco un exitoso envío de paquetes al cliente, lo cual es necesario para el streaming de video.

La pérdida de tan sólo un paquete para el streaming de video puede resultar en una pérdida o daño de todo el cuadro (frame) de video y potencialmente la propagación de un error a través de varios cuadros debido a la manera en que el video es codificado y hecho paquete. Además, la variación en el retraso de envío de paquetes, referido como jitter (fluctuación), puede también resultar en un serio “artefacto” (ruido) en el video recibido. Retardos en la transmisión dinámica puede resultar en el envío de paquetes al cliente en un diferente orden que el servidor los envió, pero el encabezamiento UDP no provee alguna información que pueda ser usada para reordenar los paquetes en el cliente. Decodificadores no suelen soportar control o técnicas de recuperación para una codificación inconsistente de una entrada de video y añadiendo pérdidas y jitter puede generar una incorrecta decodificación a la salida y posiblemente una falla al decodificar cuadros particulares.



Fig. 2.5 Impacto de errores en el streaming de video. La imagen muestra una secuencia dispareja (izquierda), con pérdida (en medio), y con reordenamiento de paquetes (derecha).

2.4.2 TCP

TCP fue diseñado para ser confiable para contrarrestar pérdida y desordenamiento de paquetes, pero todavía tiene características de transmisión que lo hacen ineficiente para un streaming de video interactivo.

TCP es otro protocolo que corre sobre IP como se muestra en la figura 2.6 pero a diferencia de UDP, es confiable porque usa retransmisión para garantizar entrega y soporta sincronización de orden de la información recibida por el cliente. Otra significativa diferencia entre estos dos, es el hecho que TCP es orientado a conexión y el estándar TCP incluye especificaciones para el control de flujo. Esto significa que el cliente y el servidor negocian para establecer una conexión para enviar información. Cada segmento TCP contiene un encabezado de al menos 20 bytes pero esto puede ser mas dependiendo en si las opciones de TCP son incluidas en el encabezado.

De la misma manera que UDP, TCP usa el número de puerto de la fuente y el destinatario para identificar el proceso de envío y recepción. El reordenamiento de información no sincronizada es logrado con TCP debido a la inclusión de un número de secuencia. La detección de paquetes perdidos o retrasados es también posible usando un número ACK (reconocimiento). El numero ACK es periódicamente enviado al servidor para revelar que paquete está esperando el cliente y usando esta información, el servidor puede decidir retransmitir cualquier paquete que pudo estar perdido en la transmisión. Estos dos campos son esenciales para que TCP sea un protocolo confiable.

El encabezado de 4 bits de largo indica el largo de los encabezados porque puede haber opciones anexas a los campos de encabezado, pero típicamente, las opciones no son usadas y este valor es de 20. A lado de las opciones, hay 6 bits de bandera en el encabezado. URG es usado para determinar si el valor en el campo de indicador urgente es válido; si lo es, el indicador urgente contiene una secuencia de número offset, el cual responde a un segmento TCP que contiene información urgente y este

deber ser enviado a su destino. ACK indica si el campo del numero ACK es válido y si lo es, este puede ser usado por el receptor para comunicarse con el servidor acerca de los paquetes que ha recibido en orden e intactos.

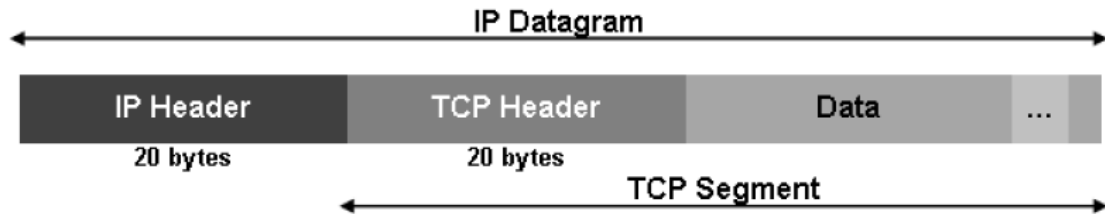


Fig. 2.6 Encapsulación de información TCP en un datagrama IP. El segmento TCP en este diagrama no incluye campos de opciones TCP.

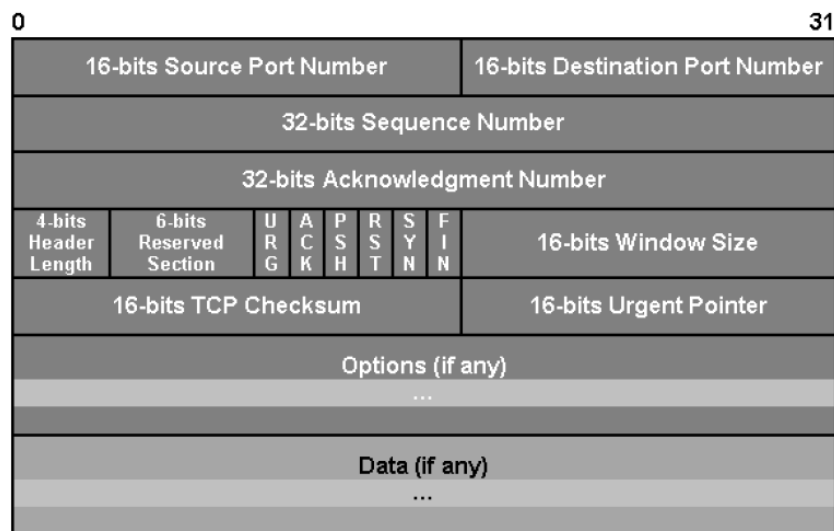


Fig. 2.7 Segmento TCP con su campo de encabezado

PSH es usado para minimizar la cantidad de buffer usado antes de enviar la información en este paquete al proceso de recepción. TCP usa un buffer en el receptor porque usa byte streams en vez de mensajes streams. La bandera RST usada para resetear la conexión, es recibida. Las banderas SYN y FIN son usadas para establecer y cerrar conexiones TCP.

El control de flujo es logrado por TCP usando el campo de tamaño de ventana. Este valor es el número de bytes, comenzando con el byte del ACK, que el receptor puede aceptar. Si el receptor está ocupado o no quiere recibir más información de la fuente,

este valor puede ser puesto a cero. La suma de control de TCP permite la detección de error en el segmento TCP.

Además del control de flujo basado en tamaño de ventana, TCP usa otro mecanismo de control de congestión como Slow Start y AIMD (Additive increase multiplicative decrease). El mecanismo de Slow Start significa que la información TCP intenta evadir la congestión, comenzando la transmisión a una velocidad baja e incrementándola gradualmente a un nivel aceptable. El AIMD significa que la velocidad de la información transmitida es incrementada lentamente mientras la red puede copiar con la velocidad actual, pero tan pronto la velocidad parece excesiva (ej: los paquetes se están perdiendo o retrasando), el transmisor baja la velocidad de la información enviada dramáticamente.

TCP es muy útil por un número posible de aplicaciones como HTTP para WWW y FTP porque estas aplicaciones no son críticas en el tiempo y la integridad de la información recibida es garantizada. Este no es el caso para streaming como audio y video que son sensibles al tiempo. El Slow Start y el AIMD no pueden ser usados en TCP para streaming de video porque necesita ser entregado en un tiempo corto. También el uso de un buffer en el receptor significa que el proceso debe esperar hasta que el buffer se llene, lo cual puede necesitar de varias retransmisiones antes de ser pasada la información. Como resultado del buffering y los mecanismos de control de flujo usados en TCP, la entrega de información al proceso de recepción no es muy consistente y esto causa una salida de video desigual.

Además a los problemas de tiempo de transmisión, TCP también introduce mucho más overhead que UDP porque debe establecer y administrar una conexión, problemas en retransmisión cuando sea requerido y su encabezado es más largo. Estos overheads pueden no ser significantes en escenarios particulares como HTTP o

FTP, pero para video el ACK, las retransmisiones y el tamaño de encabezado largo puede requerir un excesivo ancho de banda. Esto es particularmente verdadero en el contexto móvil donde el ancho de banda es un recurso escaso.

Las ventajas y desventajas de TCP y UDP para video streaming ha sido discutido, pero los dos presentan serios problemas. RTP adopta las cosas positivas de TCP y UDP y omite las que presentan problemas en streaming.

2.4.3 RTP

RTP fue diseñado para una entrega de información en tiempo real como audio y video. No es un protocolo de la capa de transporte como TCP y UDP, pero tiene características similares a un protocolo de capa de transporte. RTP usualmente corre sobre UDP y similares hacia un protocolo de capa de aplicación. Es posible usar RTP sobre TCP pero debido a los problemas de TCP, no es muy común.

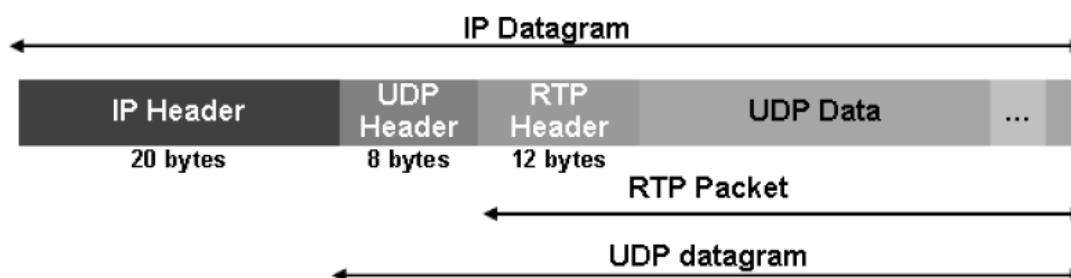


Fig. 2.8 Encapsulación de un paquete RTP dentro de un datagrama UDP e IP. Debe notarse que el encabezado RTP ilustrado no contiene ningún CSRC o encabezados de extensión

RTP es particularmente apropiado para media streaming porque permite la entrega sincronizada de la información y soporta detección de pérdida y desordenamiento de paquetes en el cliente. Ya que es basado en UDP, RTP es no orientado a conexión y el único overhead de transmisión envuelto son los encabezados IP, UDP y RTP. El encabezado RTP es de 12 bytes y los encabezados IP, UDP y RTP combinados sin

ninguna extensión adicional es de 40 bytes, equivalente a los encabezados IP y TCP en un solo paquete TCP. Los campos utilizados por RTP son mostrados en la figura 2.9.

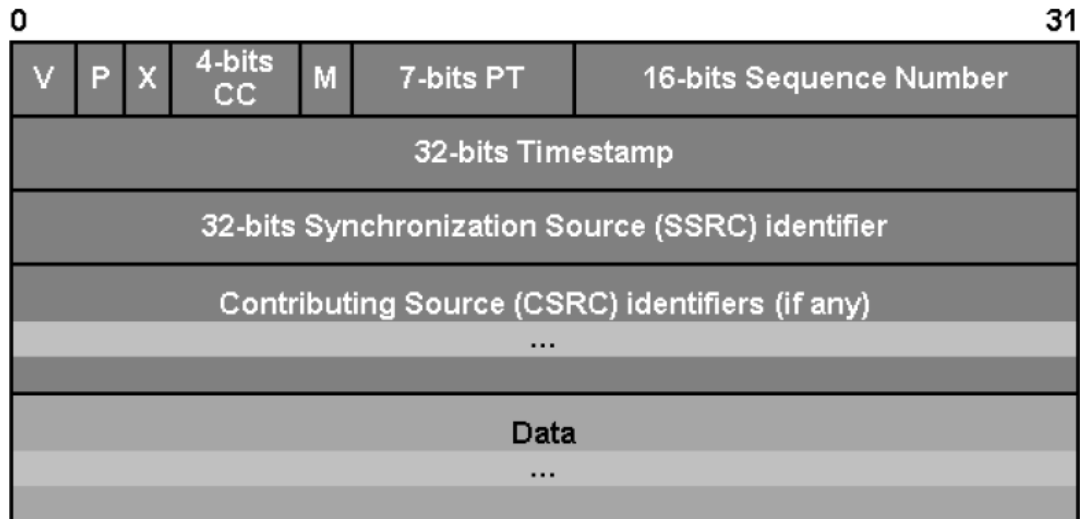


Fig. 2.9 Un paquete RTP y sus campos de encabezado

En la figura, “V” indica la versión de RTP usado y es puesto en 2 en los estándares actuales. Las banderas “P” y “X” son usadas para indicar si la información de extensión es incluida en el paquete. “CC” es el contador CSRC (Fuente de Contribución) e indica cuantos campos CSRC son anexos al encabezado RTP. CSRC son usados para identificar la fuente de la media que contribuyo para la carga de los paquetes (por ej. en una video conferencia, muchos participantes pueden contribuir con el contenido de audio).

El bit “M” puede ser usado de varias maneras definidas por un perfil, pero para video es normalmente fijado para indicar que los subsecuentes paquetes contienen el comienzo de un nuevo cuadro codificado. Una de las ideas fundamentales detrás de RTP es que cada paquete contiene información correspondiente a un tiempo exacto de media que puede ser determinado por un campo de tiempo. Sin embargo, el tamaño de carga en RTP es limitado por el MTU de la red por lo que si un cuadro

codificado es más largo que el tamaño permisible, puede ser fragmentado en dos o más paquetes RTP.

Las técnicas de codificación para los tipos diferentes de media en la carga RTP pueden variar enormemente y es hasta posible que la codificación cambie dinámicamente durante un stream RTP.

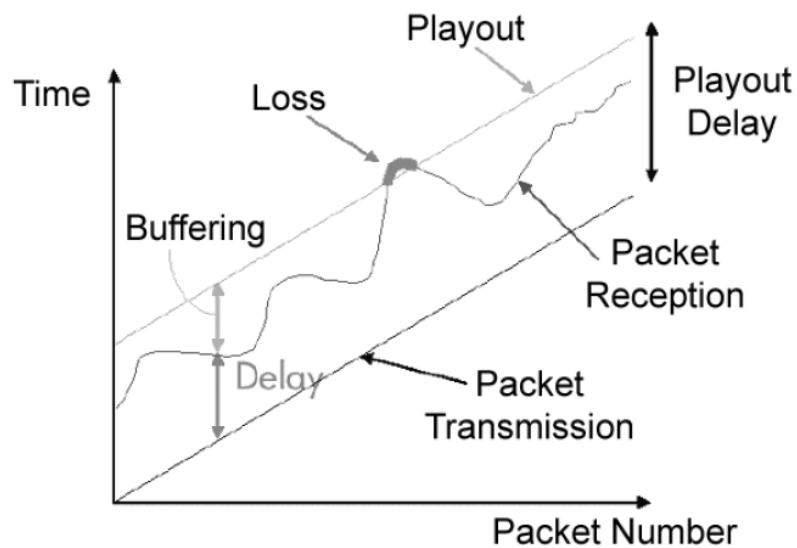


Fig. 2.10 Este gráfico muestra como el cliente de medios usa el buffering para manejar la fluctuación (jitter).

Como TCP, RTP incluye números de secuencia en el encabezado que son usados para permitir a un cliente detectar la pérdida y el desorden en la sincronización de los paquetes. Para permitir un reordenamiento y para acomodar el jitter de la red, los clientes RTP usualmente usan un buffer, pero para un media stream interactivo y en tiempo real, este buffer es pequeño. Usar un buffer pequeño significa que el jitter resultara en la entrega de los paquetes que han perdido su tiempo de reproducción. Para evitar decodificación redundante y posibles inconsistencias en el contenido decodificado, cualquier paquete que llegue tarde con respecto a su tiempo de reproducción, no son bufereados y son omitidos (es como si se hubieran perdido en la transmisión) ilustrado en la figura 2.10.

Finalmente el campo SSRC (Fuente de Sincronización) es un valor aleatorio usado para diferenciar entre streams RTP. Esto es necesario porque es posible tener más de un stream de información en el mismo tipo de carga en una sesión RTP. Una sesión RTP se refiere a todos los streams de media RTP que corresponden a una sola presentación multimedia (por ej. una sola sesión RTP puede tener un stream RTP para el video y otro para el audio). RTSP comúnmente controla un número de streams RTP en una sola sesión RTSP y cada pedido RTSP afecta todos los streams en esta sesión. El estándar RTP también incluye una especificación para un protocolo de retroalimentación llamado RTCP que permite al cliente comunicar estadísticas de recepción de paquetes al servidor. El servidor puede aplicar alguna forma de adaptación en el stream o en la calidad del video para adaptarse a las condiciones reportadas por el cliente.

RTP es más apropiado para streams de media en tiempo real porque soporta un bajo overhead de transmisión. También soporta la identificación y re sincronización de unidades de media para asegurar que el contenido válido es pasado al decodificador. Sin embargo, hay todavía un número de problemas con RTP para el contenido de video streaming especialmente en el ancho de banda para las redes móviles.

Usar RTP para una secuencia multimedia que contiene diferentes tipos de media como audio, video, texto, etc, requiere que cada stream de media sea puesto en un único stream RTP aún si el muestreo y el tiempo de todos los stream de la media están coordinados. También las técnicas de compresión de audio y video están muy avanzadas y una sola unidad de media codificada puede requerir sólo una pequeña fracción de espacio de carga disponible con RTP, particularmente cuando el video es de pequeñas dimensiones, como el usado en aparatos móviles, es transmitido. Estas

dos consideraciones pueden resultar en un uso ineficiente de ancho de banda porque una porción significativa de éste es expandido en encabezados IP, UDP y RTP.

2.4.4 RTCP

RTCP fue desarrollado para proveer mecanismos de retroalimentación para habilitar adaptaciones sobre redes IP. RTCP es basado en una transmisión periódica de paquetes de control a todos los participantes de la sesión usando el mismo mecanismo de distribución que los paquetes de información. Este protocolo debe de proveer multiplexación de la información y paquetes de control (por ej. usando números de puertos separados con UDP). RTCP realiza las siguientes cuatro funciones:

- 1) La primera función es de proveer retroalimentación en la calidad de la distribución de la información. Esta es una parte integral de RTP como un protocolo de transporte y está relacionado a la función de control de flujo y congestión de otros protocolos de transporte.



Fig. 2.11 Un paquete RTCP con sus cinco campos de cabecera

- 2) RTCP lleva un identificador de nivel de transporte para una fuente RTP llamado CNAME (Nombre Canónico). Debido a que el identificador SSRC puede cambiar y se descubre un conflicto o un programa es reiniciado, el receptor requiere que el CNAME mantenga un historial de cada participante. El receptor también requiere que el CNAME asocie múltiples stream de información de un participante dado en un conjunto de sesiones RTP relacionadas, (por ej. para sincronizar audio y video).

- 3) Las primeras dos funciones requieren que todos los participantes envíen paquetes RTCP pero la velocidad debe ser controlada para que RTP amplíe el número de participantes. Haciendo que todos los participantes envíen paquetes de control entre ellos, cada uno independientemente puede observar el número de participantes. Este número es usado para calcular la velocidad a la cual los paquetes son enviados.
- 4) Una función opcional es transportar una información de control de sesión mínima, por ejemplo identificación de participantes a ser mostrada en la interface de usuario. Esto es más útil en una sesión donde los participantes entran y salen sin control de membresía o negociación de parámetros.

Las funciones de la 1 a la 3 son obligatorias cuando RTP es usada en un ambiente multicast IP y son recomendados para todos los ambientes. Diseñadores de aplicaciones RTP son recomendados de evitar mecanismos que sólo funcionen en modo unicast.

Las transmisiones RTCP pueden ser controladas separadamente por remitentes y receptores para casos como links unidireccionales donde la retroalimentación desde los remitentes no es posible.

2.4.5 RTSP

RTSP es usado para establecer y controlar uno o más streams sincronizados de medias continuas como audio y video. Es un protocolo de la capa de aplicación y habilita control para la entrega de media en tiempo real y es compatible con contenido de media en vivo y pregrabada. Los mecanismos de control de RTSP son comparables a un control remoto de un VCR (Video Cassette Recorder) porque permite al cliente mandar mensajes al servidor para invocar a la reproducción, grabación, adelantar, rebobinar, pausar y otras operaciones específicas de la media.

RTSP define la sintaxis y los tipos de pedidos que deben ser soportados por un cliente y un servidor RTSP compatibles. Los pedidos pueden ser en cualquiera de las direcciones dependiendo del tipo de pedido. El formato de pedido es basado en HTTP/1.1, el cual a diferencia de las previas versiones de HTTP no requiere de una conexión persistente al servidor porque este servidor no puede usar la conexión para identificar diferentes sesiones, así que asigna números de sesión a cada cliente que establece una sesión RTSP con el servidor y luego todos los pedidos subsecuentes por el cliente deben incluir el número de sesión asignada.

Los diferentes pedidos pueden ser usados dependiendo de la implementación RTSP porque el estándar define un número de métodos que son recomendados u opcionales y otros que son requeridos. También es posible modificar RTSP para facilitar la funcionalidad por defecto extendiendo o alterando los parámetros del comando.

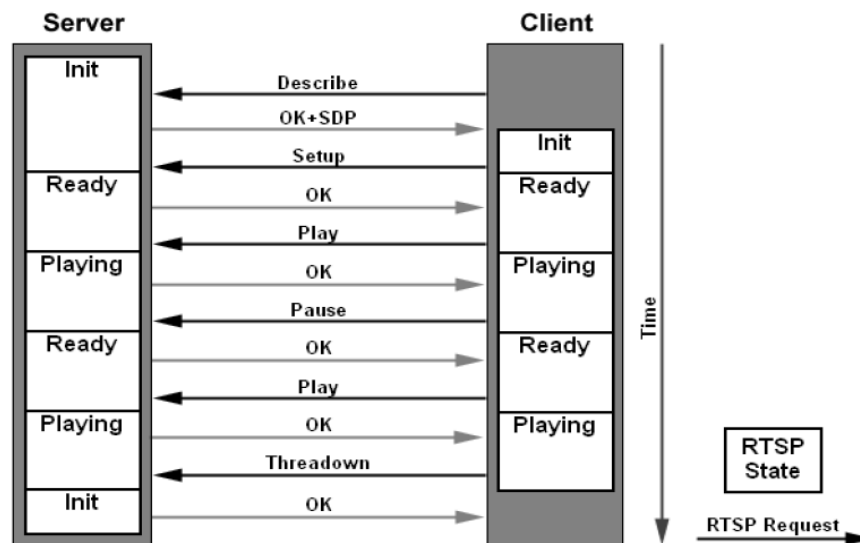


Fig. 2.12 Un ejemplo de una sesión RTSP entre servidor y cliente

En la sesión mostrada en la figura 2.10, el cliente pregunta por una descripción de algún contenido en el servidor usando el comando DESCRIBE y el servidor entrega información en formato SDP con detalles relevantes de la media consultada por el cliente. Luego el cliente llama al pedido SETUP y el servidor configura la entrega del stream para acomodarse a las opciones preferidas por el cliente. Luego PLAY es

enviado por el cliente y el servidor comienza a transmitir la información de la media. Luego un pedido de PAUSE es requerido al servidor para que deje de transmitir el stream temporalmente y el servidor espera en el estado READY para que el cliente aguarde por nuevas instrucciones. Eventualmente el cliente pide el comando PLAY y el stream se resume. Finalmente el cliente envía el pedido TEARDOWN y el servidor termina la sesión y vuelve al estado init esperando por otras sesiones a ser establecidas.

2.5 JUSTIFICACION DEL FILTRAJE VS ANALISIS DE LA FUENTE

Durante el proceso de codificado de un vídeo para terminales móviles se generan dos flujos de datos básicos, el “stream elemental” (ES) de audio y el ES de vídeo que contiene la información referente a la parte visual. En él ES de vídeo se envía una secuencia de imágenes separadas temporalmente según la tasa de frames configurada. El tamaño de cada una de las imágenes es variable dependiendo de su contenido así como del esquema de codificación. Los estándares de vídeo permiten diferentes esquemas de codificación para cada una de las imágenes:

Imágenes intra I: Las imágenes I son frames comprimidos mediante una transformada de coeficientes discretos. Una vez obtenidos estos coeficientes realizamos un proceso de cuantificación donde se filtran los coeficientes principales (coeficientes de baja frecuencia). A partir de aquí, se reordenan los coeficientes, se les aplica una codificación entrópica y obtenemos la imagen intra I. Los frames tipo I son indicados para cambios de escena y la re-sincronización del stream de vídeo, no necesitando ninguna referencia temporal anterior o posterior para poder ser decodificado.

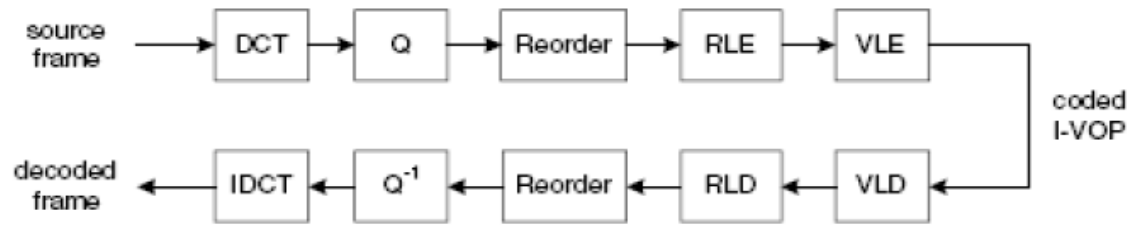


Fig. 2.13 Codificador de Imágenes I

Imágenes inter, P, B: Las imágenes tipo P son codificadas a partir de la predicción respecto al frame anterior, codificando únicamente las diferencias entre ambos. Como consecuencia la información a transmitir en el caso de una frame tipo P es menor que en los frames tipo I. Los frames B extienden la predicción, permitiendo la predicción de movimiento bi-direccional. En el caso de los frames B puede realizarse la predicción sobre frames I o frames P. En ambos casos los frames tipo I son imprescindibles para la decodificación del stream de vídeo, ya que son utilizados como imágenes de referencia.

En la siguiente figura 2.11, encontramos los bloques ME (Estimación de Movimiento) y MCP (“Predictor de Compensación de Movimiento”) encargados de predecir e identificar los movimientos experimentados por los macro bloques.

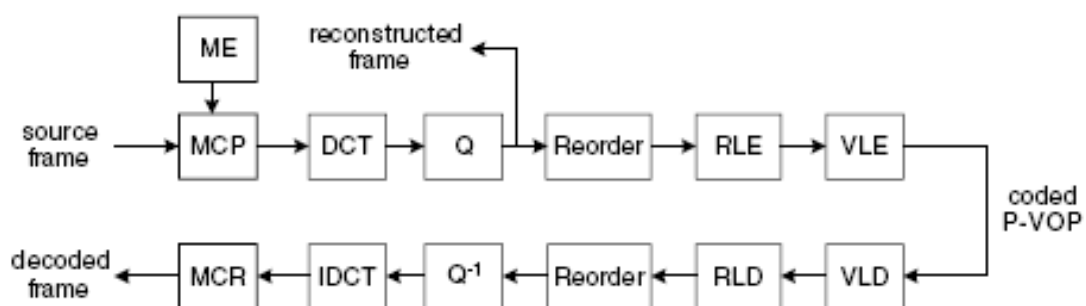


Fig. 2.14 Codificador de Imágenes P.

El grupo de frames comprendido entre dos fotogramas tipo I es conocido como GOP (group of pictures).

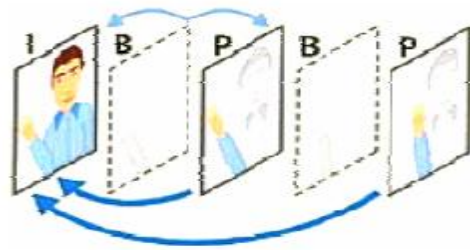


Fig. 2.15 Grupo de imágenes.

A modo de ejemplo presentamos las siguientes figuras donde se representa las ideas anteriormente presentadas.



Fig. 2.16 Imagen de referencia n



Fig. 2.17 Imagen n+1

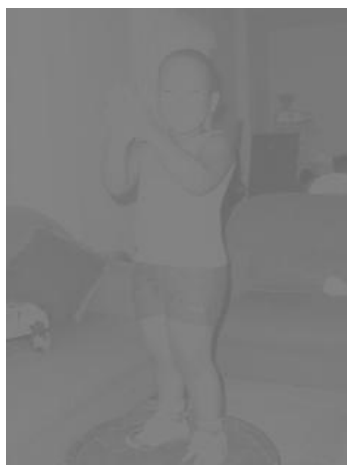


Fig. 2.18 Diferencia entre la imagen n y n+1



Fig. 2.19 Estimación de los vectores de movimiento

La figura 2.16 contiene el inicio de una secuencia de vídeo, con lo que esta imagen se consideraría frame de referencia o imagen tipo I. En la imagen siguiente se obtiene una imagen similar a la anterior, con lo que si se realiza la diferencia entre los dos frames se obtiene la figura 2.18 donde se observa una disminución considerable de la información a codificar. Finalmente en la figura 2.19 se representa la predicción de movimiento realizada para cada bloque, con lo que la imagen P está formada por la codificación entrópica de los vectores que han realizado una estimación del movimiento.

Del proceso de codificación anterior, y matizando la inexistencia de las imágenes B en los perfiles de vídeo para terminales móviles, obtenemos frames I con mucha información, junto con frames P con información referencial respecto a las I.

2.6 TIPOS DE SERVICIO

Dentro de UMTS podemos encontrar diferentes tipos de servicio, cada uno asociado a unas restricciones determinadas que nos permite establecer unos parámetros mínimos (QOS), necesarios para poder ofrecer el servicio con calidad.

Si llevamos esto a la práctica UMTS propone ofrecer mecanismos de QOS en las diferentes capas que lo componen. De esta forma para que un usuario reciba un servicio con QOS han de ponerse de acuerdo todos los elementos de UMTS, ya que entre ellos han de negociar como llevar a cabo dicha tarea. Esta característica se conoce como "End-to-End QOS " (Calidad de Servicio Extremo a Extremo).

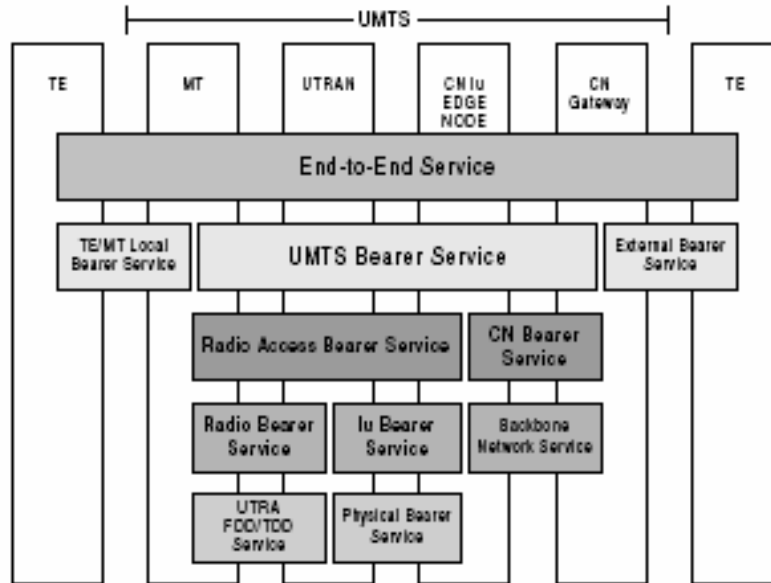


Fig. 2.20 Desglose de diferentes niveles en la escala de QoS

En el estándar podemos encontrar definidos los siguientes servicios, donde cada uno de ellos queda definido por unas características admisibles respecto a tasa de error y retardo permisible para ofrecer dicho servicio.

1. Conversacional: servicios de VOIP
2. Interactivo: juegos y aplicaciones móviles.
3. Background: servicios como WWW
4. Streaming: aplicaciones de vídeo, etc

Resumiendo las necesidades de cada servicio presentamos la siguiente figura 2.21 donde se muestra la relación entre los servicios y los parámetros deseados para cada uno de ellos.

	<i>Clases de servicio</i>			
	<i>Conversational</i>	<i>Streaming</i>	<i>Interactive</i>	<i>Background</i>
Ejemplos de servicios finales que habilitan	<ul style="list-style-type: none"> ▪ Voz conversacional ▪ Videoteléfono ▪ Telemetría/telecontrol ▪ Juegos interactivos 	<ul style="list-style-type: none"> ▪ Audio alta calidad ▪ Vídeo ▪ Transferencia de ficheros ▪ Imagen estática ▪ Telemetría (monitorizar) 	<ul style="list-style-type: none"> ▪ Mensajería de voz ▪ Navegación web ▪ Transacciones ▪ Correo electrónico 	<ul style="list-style-type: none"> ▪ Descarga de BB.DD. ▪ Descarga de correo ▪ Telemetría (telemetría) ▪ SMS
Tipos de datos	Audio/video/datos	Audio/video/datos	Audio/datos	Datos
Simetría	Bidireccional	Unidireccional	Uni y bidireccional	Unidireccional
Ancho de banda	4,25 – 384 kbps	32 – 384 kbps	4,25 – 384 kbps	Hasta 384 kbps
Retardo máximo	<200 – 400 ms	<10 s	<1 s	>10 s
Latencia máxima	Baja	Baja	Baja a media	Alta
Tolerancia a errores	Media a alta	Nula a media	Nula a alta	Nula
Tecnologías de red	GSM/UMTS	GPRS/UMTS	GPRS/UMTS	GSM/GPRS/UMTS

Fig. 2.21 Especificaciones de las necesidades de los diferentes servicios

2.7 PARAMETROS DE CALIDAD EN EL VIDEO

Para la estimación de la calidad de un vídeo existen dos modelos claramente diferenciados, los modelos basados en medidas de calidad subjetivas y los modelos basados en medidas objetivas.

Los modelos basados en medidas de calidad subjetivas se basan en la valoración por parte del usuario de una serie de componentes que proporcionan al usuario una sensación de calidad valorable en diferentes escalas. Comúnmente estos mecanismos se basan en la cuantificación de la experiencia en una escala discreta desde una calidad “excelente” hasta calidad “pobre”. Para tal efecto, dentro de las recomendaciones de la ITU encontramos unas directivas para llevar a cabo las evaluaciones asentadas en el procedimiento DSCQS (Escala de Calidad Continua de Doble Estímulo) basado en la comparación consecutiva de los elementos a evaluar.

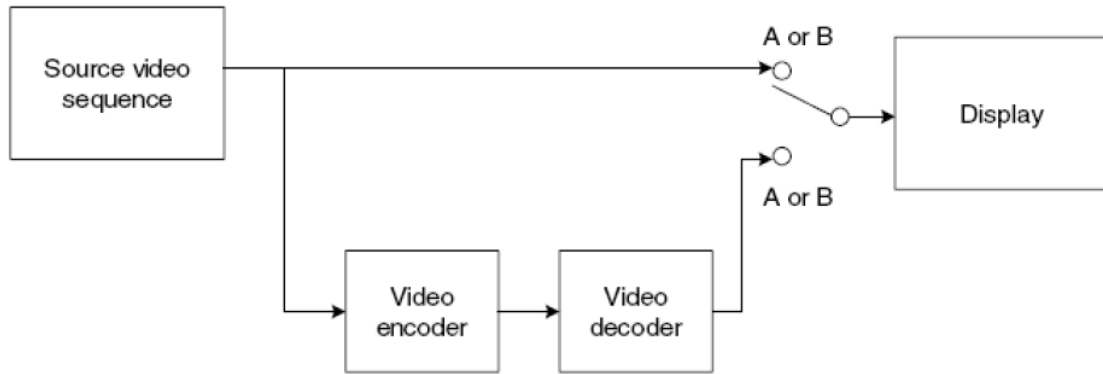


Fig. 2.22 Esquema del proceso básico para la evaluación mediante DSCQS

Por el otro lado tenemos los mecanismos basados en medidas de calidad de forma objetiva, que nos permiten automatizar las medidas usando diferentes algoritmos. El principal parámetro utilizado es el PSNR que intenta obtener una aproximación a la respuesta humana determinando un coeficiente. Dicho coeficiente se puede obtener al evaluar la siguiente expresión:

$$PSNR = 10 \log_{10} \left[\frac{(2^n - 1)^2}{MSE} \right]$$

Ecuación 2.1 Expresión para evaluar la PSNR entre imágenes

Las siglas MSE corresponden con el error medio cuadrático (Mean squared error). A modo de ejemplo presentamos las imágenes donde se ha realizado un test sobre la imagen original (a). La imagen (b) tiene una PSNR de 30,6 dB superior a 28,3 dB de la imagen (c), y por lo tanto la imagen (b) se asemeja más a la original (a).

a)

b)

c)



Fig. 2.23 Ejemplos de imágenes con diferente grado de ruido

Cabe destacar que la PSNR, en algunos casos, puede contradecirse a una valoración posterior en la calidad subjetiva, ya que se basa en la estimación media del error por cada píxel, no diferenciando si el píxel corresponde a un fondo o un contorno, donde el mismo error provocaría una peor calidad subjetiva.

Remarcar finalmente que la ITU-T ha creado un grupo de expertos para el estudio de técnicas para la valoración objetiva de imágenes y vídeos. Dicho grupo se conoce como VQEG (Grupo de Expertos de Calidad de Video) .

ESTANDARES DE CODECS DE VIDEO

3.1 CODECS UTILIZADOS

Uno de los aspectos más importantes en todo el proceso de estandarización de los servicios sobre redes móviles corresponde a la posibilidad de intercambiar información audiovisual de forma independiente al terminal móvil que se está empleando. Para ello los fabricantes de teléfonos móviles y los proveedores de contenidos han redactado una serie de normativas a través del 3GPP para definir los formatos de la información que han de soportar los dispositivos móviles. Esta normativa se encuentra recogida en un documento donde encontramos especificados los codecs a utilizar para cada servicio que se desea ofrecer. A continuación realizamos un breve resumen de los aspectos más importantes a la hora de codificar audio y vídeo.

3.2 CODECS PARA AUDIO Y VOZ

Los contenidos de audio están localizados en dos componentes principales, contenidos vocales y contenidos de audio pertenecientes a un vídeo. El 3gpp recomienda los siguientes codecs enlazados con el perfil de audio a codificar.

Codec AMR (“Adaptative Multi Rate”), para aplicaciones como VOIP donde se codifique voz.

Codec AAC (“Advance Audio Codec”), permite codificar los contenidos a tasas bajas, pero con un sonido de mayor calidad.

Esta alternativa es indicada para codificación de audio en secuencias de vídeo como vídeo clips, trailers (avances) de películas, etc.

3.3 CODECS PARA VIDEO

Para servir contenidos de vídeo sobre terminales móviles se ha consultado las especificaciones técnicas disponibles en el 3GPP, verificado con recomendaciones para desarrolladores de aplicaciones del forum Nokia.

En la siguiente tabla realizamos una comparativa de las recomendaciones obtenidas en ambas fuentes, remarcando que el documento del Nokia fórum cita en numerosas ocasiones los estándares del 3GPP.

Estándar	Códec/s	Perfil	Resolución	Bitrate	Soporte
3GPP	H.263	Perfil 0 Nivel 45	176 x 144 ¹	64 Kbps	Obligatorio
	H.263	Perfil 3 Nivel 45	-	-	Opcional
	MPEG-4	Visual Simple Perfil Nivel 0	176 x 144	64 Kbps	Opcional
	H.264	Baseline Perfil Nivel 1	-	-	Opcional
NOKIA FORUM	H.263	Perfil 0 Nivel 10	176 x 144	57 Kbps a 15 fps	Obligatorio

Tabla 3.1 Resumen de codecs soportados

Analizando el cuadro anterior podemos concluir que el códec utilizado y recomendado por ambas entidades es el H.263 perfil 0.

En la tabla anterior asociamos la palabra perfil a unas características técnicas que restringen las capacidades y técnicas de codificación a emplear en función de la finalidad del vídeo ligado directamente con las capacidades hardware de los terminales involucrados en el proceso. El nivel aporta más definición dentro de las características del perfil, estableciendo restricciones de bitrate, resolución, etc.

El perfil 0 nos define unas restricciones en cuanto a las herramientas técnicas a utilizar para la compresión del vídeo, resumiéndose en las siguientes características:

- Perfil muy sencillo y de baja complejidad.
- Posibilidad de predicción temporal mediante frames del tipo I y P, únicamente.
- No se necesitan buffer para la posible decodificación.

Todos los codecs anteriormente mencionados se basan en aprovechar modelos de predicción temporal y espacial, con lo que finalmente codifican toda la información

utilizando códigos entrópicos, reduciendo el número de bits necesarios para codificar la información.

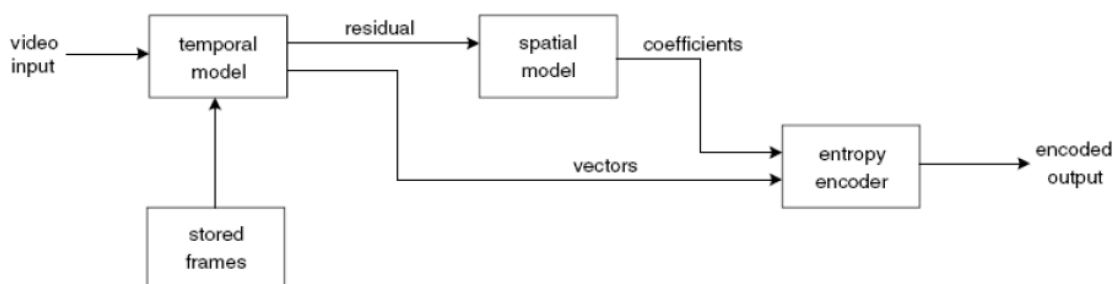


Fig. 3.1 Esquema del proceso de codificación.

Una vez que obtenemos la información de vídeo comprimida, necesitamos darle formato y ponerla en un “contenedor” para poder ser transportada y almacenada. En la actualidad uno de los formatos más conocidos es el formato AVI (“Advanced Vídeo Interleaving”). Finalmente el 3GPP ha definido un nuevo formato cuya extensión es .3gpp y es el contenedor recomendado por el mismo organismo para la entrega del vídeo a terminales móviles.

Una vez que tenemos el vídeo almacenado en un formato conocido, la aplicación servidora analiza y fragmenta el contenido para transmitir el vídeo al cliente. En este caso el protocolo utilizado es RTP, cumpliendo unas reglas de fragmentación.

3.4 ANALISIS DE CARACTERISTICAS DESDE TERMINALES ACTUALES

Otro de los aspectos importantes a tener en cuenta son las especificaciones técnicas que nos ofrecen los fabricantes de terminales para así corroborar las recomendaciones del 3GPP.

Los terminales analizados corresponden a varios modelos de las compañías Nokia y Sony-Ericson. Remarcar que los fabricantes no aportan muchos detalles técnicos de

las características de los terminales en referencia a los codecs soportados, existiendo programas auxiliares encargados de cuantificar el rendimiento de los terminales móviles.

A continuación mostramos los modelos analizados junto a sus características técnicas más relevantes.

Fabricante	Modelo	Soporte H.263	Soporte MPEG4	Audio	Otros
NOKIA	7710	Si	No	AMR	CDMA
	9500	Si	Si	AMR,AAC	CDMA
	6600	Si	Si	AMR	CDMA
	3155i	Si	Si	AMR	CDMA
Soni-Ericson	P910	Si	Si	AAC	GSM
	Z800i	Si	Si	AAC	CDMA

Tabla 3.2 Comparativa de terminales móviles

En las especificaciones citadas en los terminales anteriores no encontramos información sobre el número de frames por segundos ni el bitrate máximo del vídeo.

CAPITULO 4

IMPLEMENTACION

4.1 COMPRESION DE VIDEO UTILIZANDO SIMULINK

Esta demostración ilustra la compresión de video utilizando la transformada discreta del coseno (DCT).

La respuesta del sistema visual humano depende de la frecuencia espacial. Si pudiéramos, de algún modo descomponer una imagen en un conjunto de imágenes, cada una con una frecuencia espacial particular, podríamos separar la estructura de la imagen que el ojo puede ver a partir de la estructura que es imperceptible. La DCT puede proporcionar una buena aproximación a esta descomposición.

El esquema de simulación utilizando simulink es el siguiente:

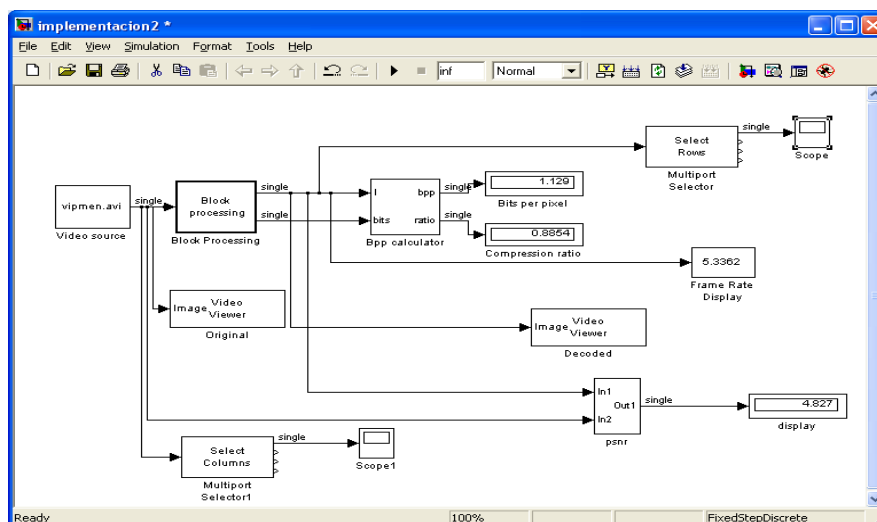


Fig. 4.1 Esquema de implementación

4.2 ADQUISICION DEL ARCHIVO DE VIDEO



Fig. 4.2 Video 1

El archivo de video.avi tiene una duración de 1 minuto 34 segundos de 640 x 480, 30 fps, tamaño 87.5 MB y velocidad de transmisión 256 Kbps. En la figura 4.2 se puede ver el archivo de video original



Fig. 4.3 Video 2 antes de la compresión

El archivo de video avi tiene una duración de 3 minuto 18 segundos de 352 x 240, 30 fps, tamaño 39.1 MB y velocidad de transmisión 256 Kbps.

4.3 TRATAMIENTO DE LA SEÑAL DE VIDEO

El bloque encargado de adaptar el archivo de video a la interface de simulink es "From Multimedia File" figura 4.3

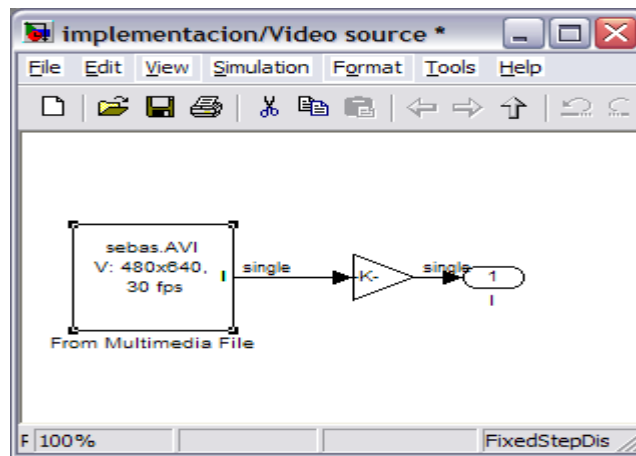


Fig. 4.4 Bloque "From Multimedia File"

4.3.1 Bloque de procesamiento

Este bloque extrae de la matriz de entrada submatrices de un tamaño especificado por el usuario. Se envía cada submatriz a un subsistema de procesamiento, y se re ensambla cada salida del subsistema en la salida de la matriz. En el bloque de procesamiento se encuentra el bloque de compresión y dentro de este el bloque que utiliza el algoritmo de la transformada del coseno (DCT) figura para realizar la compresión.

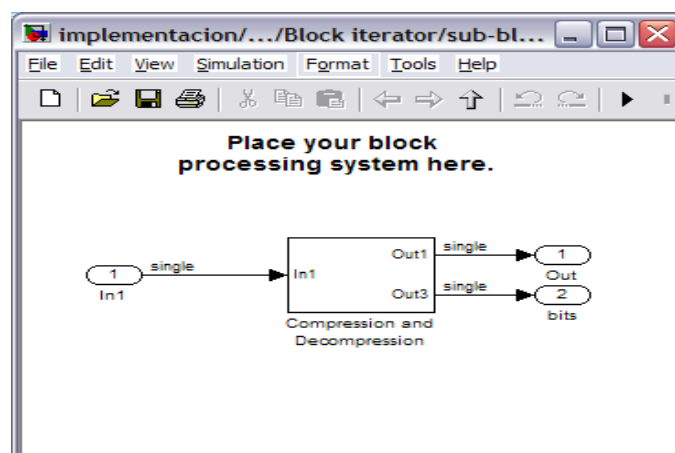


Fig. 4.5 Bloque de compresión

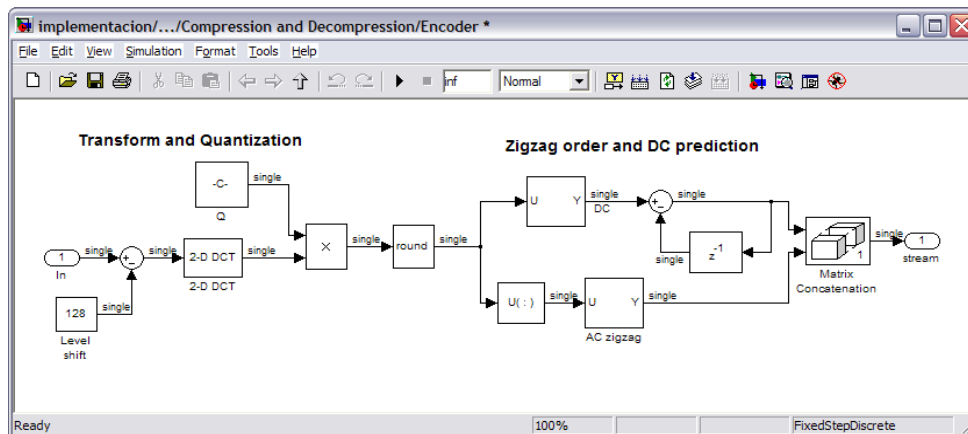


Fig. 4.6 Bloque DCT

En la figura 4.6 podemos observar el video 1 después de la compresión



Fig. 4.7 Video 1 después de la compresión



Fig. 4.8 Video 2 después de la compresión

4.4 SOFTWARE DISPONIBLE PARA TRABAJAR CON VIDEO

PARA TERMINALES MOVILES

Ahora nos concentramos en la búsqueda de software que nos permita conseguir los videos en los formatos deseados. Para tal efecto se ha realizado una búsqueda en donde nos centraremos en las siguientes características técnicas:

- Codecs que soporta.
- Resoluciones que puede tener el video de salida.
- Posibilidad de obtener la velocidad de cuadro variable.
- Licencia.

En resumen presentamos una lista de software junto con sus principales características técnicas.

Software	Códecs Video	Códecs Audio	Resoluciones	Frame Rate	Licencia
ImTOO ²	H.264, H.263+, MPEG4,	AMR, MPEG1	CIF, QCIF, SQCIF	0-30 fps	Comercial
Nokia Converter	H.263	AMR	QCIF, SQCIF	0-15 fps	Freeware
Mpeg4 for IP	H.263, MPEG4, H.264	MPEG1/2, AAC	4CIF, CIF, QCIF, SQCIF,	0-30 fps	GPL
TMPGenc	MPEG1, MPEG2	MPEG1, MPEG2	4CIF, CIF, QCIF, SQCIF,	0-30 fps	Comercial

Tabla 4.1 Comparativa de software de compresión de video.

En la figura 4.9 podemos observar el aspecto principal del programa ImTOO, el cual se ha utilizado para codificar el video empleado para las diferentes pruebas. El programa nos permite convertir el video de entrada (menú izquierdo), y re-codificarlo con los parámetros especificados (menú derecho).

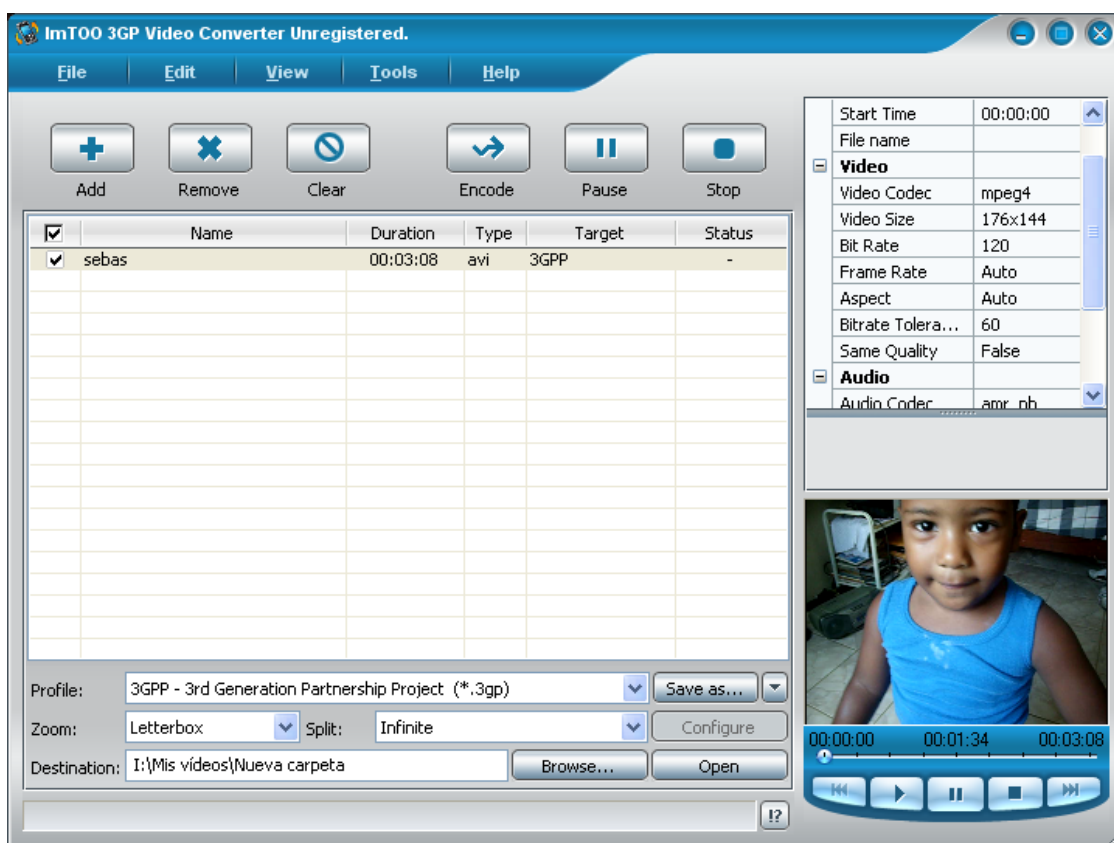


Fig. 4.9 software ImTOO para la compresión de video

4.4.1 Procedimientos para la Codificación.

Para poder comparar la calidad de los diferentes videos después de codificar, se ha optado por un punto de partida común. El punto escogido corresponde a 4 minutos de video en formato MPEG1, codificados a tasa de 8-9Mbps a 30fps. Esta fuente puede ser considerada de alta calidad, por lo que cualquier codificación tendrá como fuente de partida el citado video.

Como ya se ha mencionado el 3GPP recomienda diferentes formatos para el video, por lo que el proceso de codificación esta ajustado con los siguientes parámetros para conseguir que el video siga las recomendaciones del 3GPP.

1. Audio

- Algoritmo de compresión (MPEG, AAC)
- Velocidad de bit medio.

- Frecuencia y canales de muestreo.

2. Video

- Codec de compresión(h.263, XVID, MPEG4)
- Velocidad de bit medio.
- Cuadros por segundo del video
- Resolución

Para observar la flexibilidad que nos permiten la variación de los parámetros especificados, se presenta el siguiente apartado, donde se han realizado pruebas con diferentes combinaciones de video.

4.4.2 Ejemplos de codificación

Las pruebas realizadas consisten en re-codificar el video y el audio con diferentes codecs y obtener algunos parámetros significativos.

Para las pruebas se ha escogido un video con los siguientes aspectos técnicos:

- Secuencia de video de entropía media
- Duración: 264 segundos
- Codec de partida: MPEG1
- Tamaño:58Mb
- Resolución: 352 x 240
- Bitrate: 1850Kbps
- Cuadros por segundo: 30
- Sonido: canal (estéreo)

Para poder servir video sobre terminales móviles, y como recomienda el 3GPP, se han re-codificado los videos con H.263, MPEG4 y XVID obteniendo los siguientes resultados:

Codec	Resolución	Velocidad de bit medio Kbps	Tamaño final Bytes	Calidad MOS
<i>Xvid</i>	<i>176x144</i>	<i>40</i>	<i>1.28MB</i>	<i>4.80</i>
<i>MPEG4</i>	<i>176x144</i>	<i>40</i>	<i>1.3MB</i>	<i>4.23</i>
<i>h.263</i>	<i>176x144</i>	<i>40</i>	<i>1.45MB</i>	<i>3.78</i>
<i>Xvid</i>	<i>176x144</i>	<i>80</i>	<i>1.53MB</i>	<i>4.84</i>
<i>MPEG4</i>	<i>176x144</i>	<i>80</i>	<i>1.52MB</i>	<i>4.70</i>
<i>h.263</i>	<i>176x144</i>	<i>80</i>	<i>1.53MB</i>	<i>4.15</i>
<i>Xvid</i>	<i>128x96</i>	<i>80</i>	<i>1.53MB</i>	<i>4.75</i>
<i>MPEG4</i>	<i>176x96</i>	<i>80</i>	<i>1.51MB</i>	<i>4.54</i>
<i>h.263</i>	<i>176x96</i>	<i>80</i>	<i>1.50MB</i>	<i>3.98</i>

Tabla 4.2 Resultados de la re-codificación de la secuencia de video

Como se puede observar el tamaño del fichero ha disminuido notablemente gracias a la dalmación de resolución y la modificación de la velocidad de bit del video. A consecuencia de este fenómeno también se ha perdido calidad en el video, efecto que se manifiesta cuando visualizamos el video.

CAPITULO 5

ANALISIS DE RESULTADOS

Al comparar la señal resultante de los dos videos con su respectiva señal original se observa una reducción del nivel de detalle; esto se debe a que esta técnica de

compresión (DCT) extrae una serie de coeficientes, y al ser estos cuantificados provocan dicha reducción.

El video es segmentado en pequeños paquetes; hay paquetes que son unidos para formar de nuevo un cuadro, luego descomprimidos y guardados en un buffer; los cuadros deben estar disponibles a un proceso de visualización a una tasa precisa (por ej. un cuadro cada 33ms).

Para apreciar como la compresión de un video influye en su transporte por la red se han elegido tres parámetros, los cuales son: PSNR, Velocidad de Cuadro y Radio de Compresión.

De cada video se escogieron 3 muestras de cada parámetro. Estos valores varían debido a que en cada muestreo que realiza Simulink, se obtienen imágenes diferentes a pesar de ser el mismo video.

Al observar las muestras de PSNR, se aprecia que video posee mejor calidad. Valores de velocidad de cuadro por debajo de 20 son captados por el ojo humano como una reproducción lenta como se puede observar al reproducir el video 1 en el cual tenemos un valor promedio de 5 fps y en el video 2 un promedio de 20 fps que proporciona una reproducción "full motion" (movimiento real).

El radio de compresión nos indica cuanto ha disminuido el tamaño del video debido a la compresión realizada, y en los ejemplos se observa una reducción considerable en el tamaño de nuestros archivos de video lo que facilita su visualización en tiempo real. Aunque ahora existen codecs de compresión como el xvid mostrado en nuestro software para terminales móviles (ImTOO) que pueden comprimir video hasta en un 50% al de su tamaño original con una calidad cercana a la de la fuente.

Video 1		
PSNR	Frame Rate	Compression ratio
2.82	4.63	0.71
8.19	5.33	0.69
9.36	5.33	0.68

Tabla 5.1 Resultados de la simulación del video 1

Video 2		
PSNR	Frame Rate	Compression ratio
1.38	19.37	0.59
6.14	19.37	0.55
8.00	18.79	0.52

Tabla 5.2 Resultados de la simulación del video 2

CONCLUSIONES

1. Aunque visualmente no se observa una reducción en la calidad del video, la relación señal – ruido nos indica que existe diferencias entre el video de entrada y el comprimido, y mientras mayor sea esta relación PSNR, mejor será la calidad del video.
2. Debemos tomar en cuenta que un video es una sucesión de imágenes, por lo que, para que un video pueda ser apreciado por una persona como “movimiento completo”, este debe tener una velocidad de cuadro de 30 fps. A valores menores que 20 fps, se pierde detalle en el video.
3. El software codificador usa un algoritmo propietario para crear una forma compacta del archivo original. El codificador reduce el video reemplazando los cuadros originales con unas versiones más compactas usando algoritmos como wavelet, fractal o el DCT que fue el que se utilizó en este proyecto.
4. Al realizar la codificación variando la velocidad media de bits a la que se transmite el video, se logra una mayor rapidez en el envío de los streams, pero esto repercute en una considerable disminución de la calidad del mismo.

RECOMENDACIONES

1. Se recomienda utilizar dos videos, uno de alta calidad y otro de calidad media-baja, ya que al usar la compresión del video, no se puede apreciar a simple vista el cambio de calidad en un video de alta calidad, pero la diferencia es más notoria cuando se realiza la compresión a un video de baja calidad.
2. Debido a que un video es una secuencia de imágenes diferentes, los valores de PSNR, Velocidad de Cuadros y Compresión de Radio variarán conforme Simulink tome estos datos, por lo que se recomienda recoger varias muestras de estos factores para un correcto análisis teórico de los mismos.
3. Para calcular la PSNR con la herramienta SIMULINK de una imagen a color, es necesario separar la porción de intensidad de la imagen de la información de color.

REFERENCIAS

- [1] 3GPP TS 26.234 V5.6.0 (2003-09), Transparent end-to-end Packet-switched Streaming Service (PSS), Technical Specification Group Services and System Aspects, 2003
- [2] ITU-T Recommendation H.323, "Packet-based multimedia communication systems", <http://dret.net>, 2003.
- [3] Schulzrinne H., "Real time streaming protocol (RTSP)", IETF RFC 2326, 1998.
- [4] Goor S. A., "Experimental Performance Analysis of RTP-Based transmission Techniques for MPEG-4", IEEE Packet Video (PV04), 2004.
- [5] Schulzrinne H., "RTP: A transport protocol for real-time applications", IETF RFC 3550, 2003.
- [6] ITU-R Recommendation BT.1683, "Objective perceptual video quality measurement techniques for standard definition digital broadcast television in the presence of a full reference", <http://www.its.bldrdoc.gov>, 2003
- [7] Nokia Forum, Video Sharing , Nokia Forum, Febrero 2005.

GLOSARIO

3GPP	3rd Generation Partnership Project Asociación del Proyecto de Tercera Generación
AAC	Advanced Audio Coding Codificación de Audio Avanzado
AIMD	Additive Increase Multiplicative Decrease Incremento Aditivo / Decremento Multiplicativo
AN	Access Network Red de Acceso
AP	Access Point Punto de Acceso

BTS	Base Transceiver Station Estación Base de Transmisión
CRRM	Common Radio Resource Management Gestión de Recursos de Radio Comunes
CS	Circuit Switched Conmutación por Circuito
EDGE	Enhanced Data rates for GSM Evolution Tasa de datos Mejorada para la Evolución de GSM
E2E	End-to-End Extremo a Extremo
GERAN	GSM/EDGE RAN
GGSN	Gateway GPRS Support Node Nodo de Soporte a la Puerta de enlace GPRS
GOB	Group Of Blocks Grupo de bloques
GPRS	General Packet Radio Service Servicio General de Paquetes vía Radio
GSM	Global System for Mobile Communications Sistema Global para Comunicaciones Móviles
GTP	GPRS Tunnelling Protocol Protocolo de Tunel de GPRS
HTTP	HyperText Transfer Protocol Protocolo de Transferencia de Hiper Texto
IANA	Internet Assigned Numbers Authority Agencia de Asignación de Números de Internet
IEEE	Institute of Electrical and Electronics Engineers Instituto de Ingenieros Eléctricos y Electrónicos
IETF	Internet Engineering Task Force

	Grupo de Trabajo en Ingeniería de Internet
IMS	IP-based Multimedia Services Servicios Multi medios basados en IP
IP	Internet Protocol Protocolo de Internet
ITU-T	International Telecommunications Union - Telecommunication Standardization Sector Unión Internacional de Telecomunicaciones
LAN	Local Area Network Red de Area Local
MMS	Multimedia Messaging Service Servicio de Mensajería Multi medios
MTU	Maximum Transmission Unit Unidad de Transmisión Máxima
PDP	Packet Data Protocol Protocolo de Transferencia de Datos
PSNR	Peak Signal-to-Noise Ratio Relación Pico Señal a Ruido
QOS	Quality of Services Servicios de Calidad
RAB	Radio Access Bearer Portadora de Acceso vía Radio
RAM	Random Access Memory Memoria de Acceso Aleatorio
RAN	Radio Access Network Red de Acceso vía Radio
RSVP	Resource Reservation Protocol Protocolo de Reservación de Recursos

RTCP	Real time Transport Control Protocol Protocolo de Control de Transporte en Tiempo Real
RTP	Real-time Transport Protocol Protocolo de Transporte en Tiempo Real
RTSP	Real Time Streaming Protocol Protocolo Streaming en Tiempo Real
SDP	Session Description Protocol Protocolo de Descripción de Sesión
SIP	Session Initiation Protocol Protocolo de Inicio de Sesión
SGSN	Serving GSN
SNMP	Simple Network Management Protocol Protocolo Simple de Administración de Red
SSRC	Synchronization Source Fuente de Sincronización
TE	Terminal Equipment Equipo Terminal
TCP	Transport Control Protocol Protocolo de Control de Transporte
UDP	User Datagram Protocol Protocolo de Datagrama de Usuario
UE	User Equipment Equipo de Usuario
UMTS	Universal Mobile Telecommunications Systems Sistema Universal de Telecomunicaciones Móviles
URI	Uniform Resource Identifier Identificador Uniforme de Recursos
USB	Universal Serial Bus

	Bus Serial Universal
USIM	User Services Identity Module Módulo Identificador de Servicios de Usuario
UTRAN	UMTS Radio Access Network Red de Acceso vía Radio UMTS
VCR	Video Cassette Recorder Grabador de Cassettes de Video
WAP	Wireless Application Protocol Protocolo de Aplicación de redes sin cable (Wireless)
WLAN	Wireless LAN
WWW	World Wide Web