

SISTEMA SOFTWARE BASADO EN LA RECOMENDACIÓN H.261 PARA LA TRANSMISIÓN DE VÍDEO SOBRE IP

Jose Luis Vázquez García

Fernando Martín Rodríguez

Francisco Parada Loira

Departamento de Tecnologías de las Comunicaciones
Universidad de Vigo

jvazquez@tsc.uvigo.es

fmartin@tsc.uvigo.es

fparada@tsc.uvigo.es

RESUMEN

En esta comunicación describimos un nuevo sistema 100% software para la transmisión de vídeo por internet. El sistema se ejecuta en el sistema operativo linux y se basa en enviar tramas construidas según el estándar H.261 [1] usando el protocolo TCP.

El sistema usa algoritmos rápidos para los cálculos de la transformada DCT (Discrete Cosine Transform) y DCT inversa así como una librería desarrollada anteriormente en nuestro grupo de investigación para la programación sencilla de aplicaciones TCP/IP (REDFACIL [2]).

1. INTRODUCCIÓN

Nuestro sistema está formado por dos aplicaciones informáticas:

- **Servidor:** es un programa codificador que funciona en una máquina que dispone de tarjeta capturadora de vídeo (frame-grabber) o elemento similar (vale cualquier dispositivo de captura compatible con linux). Si no se dispone de dispositivo de captura se puede simular con secuencias grabadas en disco.
- **Cliente:** es el programa decodificador. Al arrancarlo se le especifica la dirección IP del servidor y los parámetros de la comunicación; a saber: ancho de banda máximo que se puede ocupar, formato CIF o QCIF (pantalla completa o cuarto de pantalla), y si se desea visualizar la secuencia o guardarla en el disco duro. El cliente realizará al servidor la petición de establecimiento de un flujo H.261 con los parámetros indicados. Si el servidor está disponible la conexión se establecerá y la secuencia aparecerá en la pantalla o se grabará a disco (según especificación del usuario).

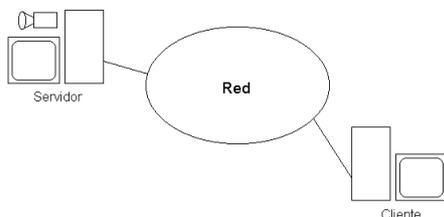


Figura 1. Esquema del sistema.

La estructura del resto del artículo es como sigue: en el apartado de descripción del sistema se describen los procesos de captura, codificación y decodificación, envío a través de TCP y visualización; después se continúa con los apartados de resultados, conclusiones y líneas futuras.

2. DESCRIPCIÓN DEL SISTEMA

2.1. Captura de Imagen

La captura se realiza en uno de los formatos especificados en la recomendación H.261 (CIF o QCIF), el que haya especificado el usuario. El número de imágenes por segundo se determina en tiempo real a partir del máximo ancho de banda disponible que se nos haya asignado (ver sección 2.3).

La captura se realiza mediante un proceso paralelo (*thread*) que se comunica con el resto del sistema.

2.2. Codificación y Decodificación

La codificación y decodificación siguen el esquema básico del estándar H.261 (ver figura 2). Donde hemos eliminado la compensación de movimiento (en el estándar no es obligatoria y es el proceso que más consumo de CPU realiza).

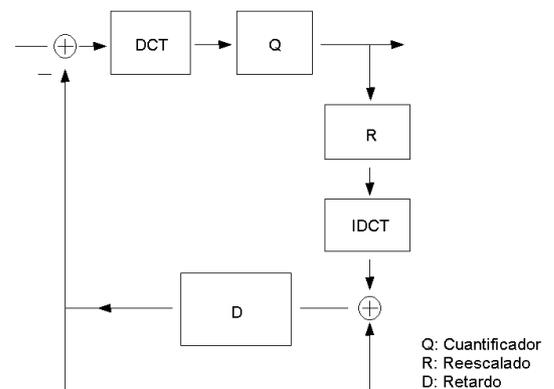


Figura 2. Esquema del codificador.

La parte más importante es el cálculo de las transformadas del coseno (DCT, tamaño 8x8) directa e inversa. Si este cálculo se realizase directamente usando la definición sería preciso realizar 8384 productos y 4096 sumas. Para evitar esta carga computacional hemos recurrido al algoritmo de Feig y Winograd [3]. Este método permite realizar una transformada coseno unidimensional con 14 productos y 35 sumas. La transformada coseno bidimensional:

$$Y = CXC^T \quad (1)$$

Se puede implementar realizando primero 8 transformadas unidimensionales:

$$\Omega = CX \quad (2)$$

Y después otras 8 también unidimensionales:

$$Y = \Omega C^T = (C\Omega)^T \quad (3)$$

Con esto se puede realizar la DCT bidimensional completa con 16 transformadas unidimensionales: 224 productos y 560 sumas. La velocidad de la anterior operación se mejora todavía más realizándola en punto fijo.

2.3. Transmisión

El acceso al protocolo TCP se ha hecho a través de la librería REDFACIL [2] que permite usar cualquier tipo de sockets en UNIX o en WIN32 con una librería de clases en C++.

Según establece el estándar, debe existir un buffer de transmisión que permita mantener el flujo de bits a una velocidad constante. Si el buffer amenaza con vaciarse (o llenarse) se varía la cuantificación para aumentar (disminuir) la calidad y aumentar (disminuir) el tamaño de las imágenes comprimidas, eso hará que el buffer recupere la ocupación óptima (la media). Enviando por TCP la velocidad binaria de salida no es tan rígida como, por ejemplo, en una línea RDSI (que es para lo que se pensó el estándar). Lo que sí es muy interesante es asegurar que el sistema no ocupe más ancho de banda que el que el usuario le especifique. Nosotros usamos el buffer de esta forma, vaciándolo (enviando) con un *thread* a velocidad constante y actuando sobre la cuantificación y también sobre el número de imágenes por segundo para mantener el buffer en ocupación media.

2.4. Visualización

La visualización se ha implementado en X-Windows usando la librería SDL. Este proceso está implementado en un *thread* separado del decodificador.

3. RESULTADOS

Hemos calculado la relación señal ruido para una secuencia de 50 imágenes obteniendo un valor medio de 32 dB para el cuantificador 1 (máxima calidad) y 14 db para el 25 (mínima calidad). Puede verse un ejemplo en la figura 3.

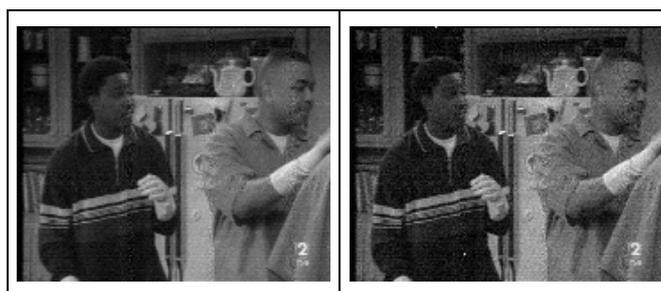


Figura 3. Imágenes de ejemplo: original (izquierda) y comprimida (derecha).

Fijando la velocidad a 64 Kbits/s el sistema ha sido capaz de enviar 2 imágenes por segundo con el cuantificador 1 (máxima calidad) y 14 imágenes por segundo para el 25 (calidad baja). Las pruebas se realizaron con un AMD K6 a 300 MHz. Además, se comprobó con la herramienta *top* que el porcentaje de CPU que ocupaba el codificador nunca pasó del 6%.

4. CONCLUSIONES

Se ha realizado un sistema sencillo pero útil que envía vídeo codificado según H.261 usando el protocolo TCP. El sistema funciona bajo linux y tiene resultados aceptables para aplicaciones de videoconferencia, videovigilancia...

5. LÍNEAS FUTURAS

Las líneas futuras de trabajo más interesantes serían:

- La realización de una versión para WIN32.
- La inclusión de audio comprimido en el sistema.
- Encontrar métodos para poder enviar el mismo flujo a diferentes receptores (multicast).

6. REFERENCIAS

- [1] CCITT. Códec vídeo para servicios audiovisuales a px64 Kbits/s. Recomendación h.261.
- [2] M. A. Piñeiro Martínez, X. Fernández Hermida. Biblioteca de funciones multiplataforma para comunicaciones TCP/IP, Proyecto fin de carrera. Universidad de Vigo. 1999.
- [3] E. Feig, S. Winograd. Fast algorithms for the discrete cosine transform. IEEE Transactions on Signal Processing, 1992.
- [4] Biblioteca SDL. <http://www.libsdl.org>.