

# Design of a scalable system for the sensorization of virtual tv sets with live broadcast

R. Méndez<sup>1</sup>, G. Sanmartín<sup>1</sup>, D. Mera<sup>1</sup>, J. Flores<sup>1</sup>, E. Castelló<sup>2</sup>

<sup>1</sup> CITIUS, USC, España <sup>2</sup> Facultade de ciencias da comunicación, USC, España

---

## Abstract

*The sensorization of virtual tv sets using motion capture systems allows the automation of many of the processes implied in the live broadcast of a tv show. It facilitates the tasks of both the actor (allowing him to interact with the environment) and the technicians (camera position and orientation, real objects position, etc.).*

*Adding new sensors usually derives in an increase of the load of the render computer and therefore a decrease in the frame rate. This increase in the load is due to the captured data processing and its application to the scene, that must be computed in parallel to the render by the same computer. This article presents a centralized hardware and software architecture that allows the addition of an unlimited number of sensors without directly affecting the frame rate of the render. The system centralizes in a single PC all the data captured by the different sensors, giving a suitable format to this data and updating the appropriate variables across the network transparently to the rendering program, keeping a stable and unaffected render time.*

Categories and Subject Descriptors (according to ACM CCS): I.3.1 [Computer Graphics]: Hardware architecture—Input devices

---

## 1. Introducción

Un escenario virtual de televisión consiste en una serie de elementos no reales generados por ordenador que se combinan con personas y otros elementos reales capturados por una cámara de video y cuya unión pretende dar una sensación de uniformidad y realismo. En estos escenarios se substituyen los elementos del plató tradicional (paredes, techos, etc.) por elementos virtuales, reduciendo así los costes de creación de un nuevo plató y aumentando las posibilidades creativas del mismo.

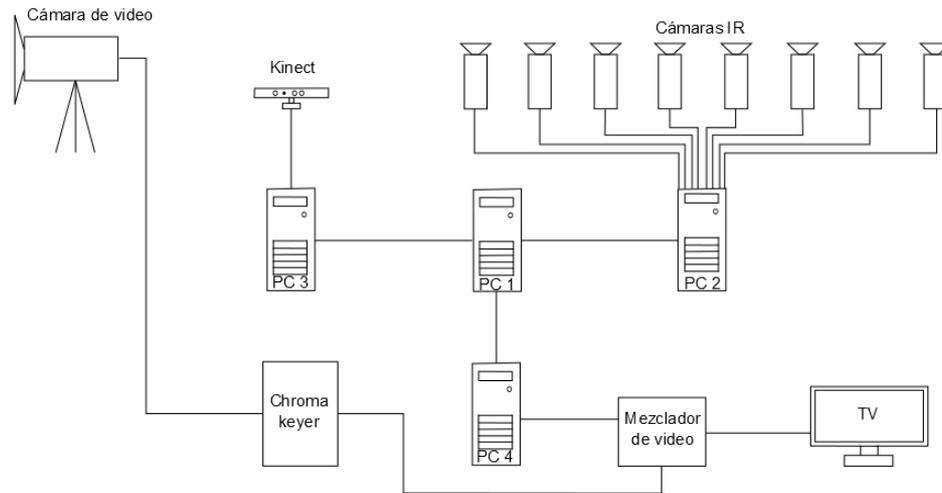
Esta combinación de mundo real y virtual se consigue a través del *chroma key*, una clave de color (un solo color primario o un patrón de varios colores [YKMM04]) que cubre todas las paredes y algunos elementos del plató virtual y que nos permite, a partir de la diferencia cromática, separar los elementos reales que se quieren conservar en la imagen final de los han de ser eliminados.

El uso de este tipo de tecnología presenta numerosas ventajas, como las enormes posibilidades de creación que tienen los productores y diseñadores al no estar limitados a la creación de elementos físicos (con las limitaciones físicas

que estos tienen), la posibilidad de que un solo plató real pueda ser usado para la emisión de diferentes programas cambiando el plató virtual o la reducción de costes al generar un nuevo plató, pues simplemente ha de crearse el modelo 3D y, si se desea, añadir una serie de elementos reales.

A pesar de estas ventajas, el uso de un plató virtual de televisión tiene también inconvenientes, como el posicionamiento de las cámaras virtuales respecto a las reales (para que la imagen del plató virtual sea coherente con la de los elementos reales gravados por la cámara real, la escena debe renderizarse utilizando una cámara virtual situada en la misma posición y orientación que la cámara real), la interacción del presentador con su entorno virtual (el presentador actúa en un espacio prácticamente vacío, sin referencias al mundo virtual y sin poder interactuar con este), o la iluminación del mundo real y el virtual, que han de ser coherentes para que la imagen final sea creíble.

Si se trata de una emisión en diferido estos problemas pueden ser solucionados en postproducción, pero cuando la emisión se realiza en directo (como es habitual en televisión) la respuesta del sistema ha de ser inmediata y por tanto los



**Figure 1:** Configuración propuesta para el sistema

problemas antes planteados deben de ser solucionados en tiempo real.

En este artículo se presenta una configuración hardware y software para la automatización de estos procesos de producción mediante el uso de sensores ópticos y un proyecto de más largo recorrido estudiando las aplicaciones de esta y otras posibles sensorizaciones para solucionar los diferentes problemas que presenta la utilización de platós virtuales de televisión a día de hoy.

En un plató virtual de televisión resulta útil una herramienta que permita realizar tracking de objetos y personas, puesto que nos permite conocer la posición y orientación tanto de las cámaras reales (dato imprescindible para el render del escenario virtual) como la posición de objetos y personas reales situados en el plató. Esto permite la interacción entre elementos virtuales y reales y la creación de feedback para el presentador [Tho06].

Para obtener estos datos en tiempo real se utilizan diferentes técnicas como: patrones en las paredes respetando el color del *chroma key* [XDD01] que únicamente permiten saber la posición de las cámaras respecto al escenario virtual, marcadores situados en el techo del estudio que permitan obtener la posición de la cámara a partir de una imagen obtenida de ellos por otra cámara perpendicular a la que está grabando [RP08], o marcadores de realidad aumentada (patrones que permiten a partir de una sola imagen obtener su posición 3D a partir de su tamaño y forma característica) que el presentador pueda manipular, modificando consecuentemente objetos virtuales como libros [BKP01], tablas de datos, etc. Combinando estas diferentes opciones se puede obtener un tracking de la cámara y de los objetos deseados, pero no se podría obtener la posición del presentador a menos que se llevase un marcador visible.

Otra opción es el uso de cámaras de infrarrojos externas que permitan hacer un seguimiento de marcadores activos (emiten luz infrarroja) o pasivos (reflejan luz infrarroja) adheridos al objeto del tracking, como la cámara, el presentador, un objeto real, etc. Aunque el escalado de estos sistemas es más costoso que el de los anteriores (pues supone aumentar el número de cámaras) la flexibilidad que ofrece al poder determinar la posición de cualquier elemento presente en el plató con solo unirle un patrón justifica, junto con su precisión, el uso de este tipo de sistemas para platós de superficie media.

Otro tipo de sensor de reciente aparición como Kinect tiene también su aplicación en el campo de los platós virtuales. La capacidad de este sensor para realizar una captura de movimiento parcial o completa del cuerpo humano en tiempo real y sin marcadores de ningún tipo puede tener numerosas aplicaciones en el campo de la interacción del presentador con su entorno virtual. Teniendo en cuenta que la precisión de este sensor no es la suficiente para la realización de tareas que necesiten una información con un gran nivel de detalle, si que se puede utilizar para otras menos complejas como detectar la posición del presentador en el espacio (sabiendo la posición del kinect), saber una aproximación de la postura del presentador para proyectar sombras, etc.

Nuestra propuesta es el uso combinado de dos tecnologías para captura de movimiento basadas en imagen: Cámaras de infrarrojos con marcadores pasivos y el sensor Kinect.

## 2. Arquitectura del sistema

Se propone un diseño flexible basado en conexiones de red entre los diferentes elementos (Figura 1), lo que permite

distribuir los cálculos necesarios en diferentes máquinas obteniendo un rendimiento mayor, o aglutinarlos en una ahorrando costes de hardware pero perdiendo en *frames* por segundo. Del plató real parten tres señales:

- La imagen de la cámara de video que graba la escena (Figura 1 Cámara de video). Este estream de video va directo al chroma keyer que separa la imagen que se conservará del chroma que será eliminado y posteriormente lo envía al mezclador de video, en el que se unirá con la imagen generada por ordenador.
- La información del sensor Kinect. Es recibida por un PC (Figura 1 PC 3) que transforma los datos y los envía a un segundo PC (Figura 1 PC 1) que centraliza toda la información antes de conectarla con el programa que realiza el render.
- La información de las cámaras de infrarrojos. Es recibida por un PC (Figura 1 PC 2), en el que un software interpreta las diferentes imágenes recibidas y obtiene las posiciones y orientaciones de los diferentes sólidos rígidos definidos en la escena. Esta información es enviada al PC en el que se centralizan todos los datos de la captura (Figura 1 PC 1) para ser enviada al programa de renderizado (Figura 1 PC 4).

El PC que hace la función de puente entre el programa de renderizado y los diferentes programas de captura de movimiento (Figura 1 PC 1) recibe los datos de un *frame* y los empaqueta según la configuración deseada por el usuario para su transmisión al programa que se encargará del render del plató virtual (Figura 3). Esta configuración, que centraliza todos los datos en un PC antes de realizar el envío permite que se puedan añadir nuevos módulos y sensores sin afectar directamente a la comunicación con el programa de renderizado. Además liberamos de cálculos al programa de render, que solo tiene que recibir los datos y aplicarlos a los objetos linkados con cada posición, permitiendo de esta forma mantener el *frame rate* necesario para una emisión en directo.



Figure 2: Marcadores pasivos unidos a una cámara

Una vez estos datos empaquetados son enviados al programa de renderizado, este actualiza la posición de la cámara y de los objetos linkados a los datos recibidos y renderiza la imagen, enviándola al mezclador de video para su composición con el video real y su broadcast.

### 3. Implementación

Se ha implementado como primera parte del proyecto y fase de pruebas inicial un sistema de captura de movimiento basado en 8 cámaras de infrarrojos y marcadores pasivos. El software utilizado en este caso es Tracking Tools, que nos permite la creación de sólidos rígidos (conjuntos de tres o más marcadores) y su posterior seguimiento en el espacio (obteniendo posición y orientación), así como el seguimiento de la posición de marcadores individuales (obteniendo solo su posición). Para el renderizado en tiempo real del plató virtual se ha empleado el software eStudio de la empresa Brainstorm Multimedia.

Para conectar estas dos herramientas se ha desarrollado un software puente, que modifica y da formato a los datos brutos obtenidos de la captura y, antes de su envío al programa de render, les aplica un filtro para evitar el ruido propio de este tipo de capturas. Tras este suavizado se empaqueta toda la información recibida en un formato adecuado para la transmisión de datos a eStudio y se envía a través de la red.

El diseño previo realizado para esta configuración nos permite una flexibilidad total a la hora de realizar la instalación teniendo como único elemento limitante la potencia de cálculo de los PCs empleados. En nuestra configuración, para estas pruebas, el ordenador centralizador de la información es el mismo que el que recibe la

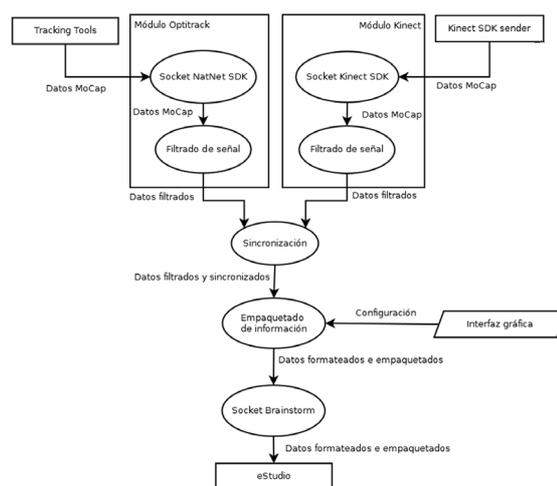


Figure 3: Flujo de datos del sistema

información de las cámaras y la procesa, ya que tiene suficiente capacidad de cálculo para realizar estas dos tareas. Podría incluso ser el mismo que realiza el render de la escena, aglutinando todo el diseño en un solo ordenador, pero la alta carga de cálculo que requiere la obtención de imágenes sintéticas y la necesidad de mantener un *frame rate* estable para la emisión en directo desaconsejan esta opción.

El hecho de añadir nuevos sensores al plató únicamente repercutiría en abrir nuevas conexiones con el PC que centraliza toda la información de captura, activar los módulos correspondientes en el software implementado y vincular esta información a eStudio. Demostrándose de esta forma la escalabilidad del sistema y su sencilla ampliación para futuras mejoras del plató.



**Figure 4:** Control del plató. Se pueden ver el render (pantalla inferior) y la unión final de render e imagen real (pantalla superior)

Se han realizado pruebas con el sensor Kinect, implementando la conexión para el envío de datos a través de la red, así como la aplicación de estos datos a un modelo simple para la generación de sombras empleando una implementación en Python y GLSL de los Variance Shadow Maps [DL06] para la obtención de *soft shadows*. Con esta implementación se obtienen 40 fps de media en una Nvidia gforce 8400 GS. Esto permite estimar como viable el uso de este algoritmo para su implantación en un plató de televisión, ya que en estos se dispone de tarjetas gráficas más potentes que la empleada para las pruebas. Esta implantación permitiría que el presentador real proyecte sombras sobre objetos virtuales, aumentando de esta forma su sensación de presencia en el mundo virtual.

#### 4. Resultados

Se ha probado el sistema para el tracking de la posición y orientación de las dos cámaras con las que cuenta el plató (Figura 2), vinculando la información de estas cámaras a las cámaras virtuales desde las que se realizará el render. Se ha observado que el *frame rate* se mantiene estable en el render y no varía respecto al de una cámara fija, puesto que

no ha de realizar ningún cálculo adicional. Al ser mayor el *frame rate* de las cámaras de infrarrojos que el de la cámara de video se evitan saltos y "tirones" en la imagen, siendo el desplazamiento continuo y suave (Figura 4).

El filtro de señal implementado asegura una mayor estabilidad de los ángulos de orientación, manteniendo fija la cámara y evitando temblores indeseados en la imagen final debidos al ruido de la señal.

#### 5. Conclusiones y trabajo futuro

Se ha diseñado una configuración de equipos flexible y escalable para la inclusión de diferentes tipos de sensores en un plató virtual. Se ha probado con éxito la implementación de este diseño y su flexibilidad al utilizarlo con herramientas comerciales y un software intermedio modular desarrollado específicamente para este diseño que permite la conexión de diferentes tipos de sensores, simplemente desarrollando y acoplando un módulo específico, sin variar el resto de la configuración del plató.

Como trabajo futuro se plantea la inclusión de nuevos sensores en la arquitectura ya definida del sistema. El primer sensor que será integrado es Kinect. Una vez que la información de este sensor sea transmitida al programa de render se utilizará para solucionar diferentes problemas de los platós de televisión actuales: ocultación de elementos reales y virtuales (saber si el presentador está delante o detrás de un elemento virtual en cada momento), sombras (integración de los shaders desarrollados en eStudio), etc.

Finalmente se probará todo el sistema con la emisión y grabación de un programa en directo utilizando todas las herramientas implementadas.

#### References

- [BKP01] BILLINGHURST M., KATO H., POUPYREV I.: The magicbook: a transitional ar interface. *Computers & Graphics* 25 (2001), 745–753. 2
- [DL06] DONNELLY W., LAURITZEN A.: Variance shadow maps. In *Proceedings of the 2006 symposium on Interactive 3D graphics and games* (New York, NY, USA, 2006), I3D '06, ACM, pp. 161–165. 4
- [RP08] RAHBAR K., POURREZA H. R.: Inside looking out camera pose estimation for virtual studio. *Graph. Models* 70 (July 2008), 57–75. 2
- [Tho06] THOMAS G.: Mixed reality techniques for tv and their application for on-set and pre-visualization in film production. *International Workshop on Mixed Reality Technology for Film-making* (2006). 2
- [XDD01] XIROUHAKIS Y., DROSOPOULOS A., DELOPOULOS A.: Efficient optical camera tracking in virtual sets. *Image Processing, IEEE Transactions on* 10, 4 (apr 2001), 609–622. 2
- [YKMM04] YAMASHITA A., KANEKO T., MATSUSHITA S., MIURA K. T.: Region extraction with chromakey using stripe backgrounds. *IEICE Transactions* 87-D, 1 (2004), 66–73. 1