UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR

Grado en Ingeniería de tecnologías y servicios de telecomunicación

TRABAJO FIN DE GRADO

ANÁLISIS Y MEJORA DE UN MODELO DE CALIDAD DE IMÁGENES VIRTUALES PARA SISTEMAS DE VIDEO MULTIVISTA

Luis Miguel Martínez Antolín
Tutor: Pablo Carballeira López
Ponente: José María Martínez Sánchez

Junio 2018

ANÁLISIS Y MEJORA DE UN MODELO DE CALIDAD DE IMÁGENES VIRTUALES PARA SISTEMAS DE VIDEO MULTIVISTA

Luis Miguel Martínez Antolín

TUTOR: Pablo Carballeira López

PONENTE: José María Martínez Sánchez

Video Processing and Understanding Lab
Departamento de Tecnología Electrónica y de las Comunicaciones
Escuela Politécnica Superior
Universidad Autónoma de Madrid
June 2018

Resumen

Las tecnologías de video 3D están avanzando mucho en los últimos años, siendo cada vez mayor el abanico de posibilidades que ofrecen y más económicas sus plataformas de visualización. Por el contrario, las pantallas estereoscópicas no han terminado de asentarse en el mercado y en los hogares de los usuarios. Problemas en la visualización como la fatiga visual o la necesidad de utilizar unas gafas para poder disfrutar del contenido son algunas de sus causas. Como solución, cada día se están mejorando los sistemas de video autoestereoscópico, es decir, de visualización 3D sin gafas, como es el caso del Super Multiview Video (SMV). El SMV es un sistema de video multivista 3D que permite al usuario ver la escena desde distintos puntos de vista. Sin embargo, este tipo de video es muy costoso de producir, tanto económica como computacionalmente, ya que será necesaria la captura, codificación y transmisión de una cantidad de vistas muy grande. Como solución a este problema, se propone la síntesis de vistas virtuales a partir de otras reales de la misma escena, de manera que se pueda prescindir de algunas cámaras y se genere el mismo contenido ahorrando en la cantidad de información a transmitir. Sin embargo, la síntesis de estas vistas puede dar problemas en la visualización, ya que se pueden generar artefactos (errores) en las vistas sintetizadas que empeoran la percepción. Para solucionar estos problemas se proponen modelos que permiten predecir la calidad del video sintetizado a partir de características de la escena, como es la información de profundidad o la distancia focal. Uno de estos modelos es el MVPDM (Multiview Perceptual Disparity Model), que propone la predicción de determinadas medidas de calidad, objetivas o subjetivas a partir de la disparidad perceptual, un parámetro que modela la percepción subjetiva del usuario a partir de la disparidad de la escena. Basado en este modelo tenemos trabajos que se encargan de realizar una predicción de la calidad de la imagen en vistas virtuales a partir de ésta disparidad perceptual. Estos modelos de predicción de calidad han sido probados para una cantidad de datos limitada, obteniendo buenos resultados. Nuestro objetivo en este TFG será el de aumentar los datos de entrenamiento de este modelo, obteniendo una mayor cantidad y variedad de los datos para ver cómo generaliza el modelo para estos nuevos datos. En primer lugar, hemos aumentado los datos en el espacio y en el tiempo, y hemos visualizado la variación del PSNR y disparidad perceptual en cada uno. Después, hemos realizado un análisis similar al del trabajo anterior para los nuevos datos de entrenamiento, y hemos comparado los resultados con los del trabajo previo, observando resultados similares y concluyendo que el modelo es genérico para una cantidad de datos mayor y más variada.

Palabras clave

Video 3D, Free Viewpoint Video, Super Multiview Video, multivista, Peak Signal to Noise Ratio, PSNR, Multiview Perceptual Disparity Model, MVPDM.

Abstract

During the last couple of years 3D Video Technologies have been advancing, offering a broader range of possibilities and more economic visualization platforms. On the contrary, 3D video is a content that has not finished settling in the market and in the users' households. Some of the causes to this issue are visual fatigue or the necessity of goggles to enjoy the video content. As a solution, autostereoscopic video systems, that is, 3D viewing systems that do not require goggles, like the Super Multiview Display (SMV) are being improved every day. The SMV is a 3D Multiview video system that allows the user to watch the setting from different points of view. In some cases, it even allows the user to choose the point of view he or she wishes to watch the scene from. However, since there is a need to capture, code and transmit a vast amount of views, this type of video is hard to manufacture both economically and computationally. As a solution to this problem, it is proposed that the virtual views are synthesized based on other real ones from the same setting. By doing this, we get rid of some cameras and transmit less information but still produce the same content. However, the synthesis of this views can generate problems when visualizing them due to the fact that they can create errors in the synthesized views which deteriorate the overall perception. As a solution to these problems, models that can predict the quality of the synthesized video based on characteristics of the scene such as the depth of field or the aperture are proposed in this paper. One of these models is the MVPDM (Multiview Perceptual Disparity Model). This model proposes the prediction from the perceptual disparity of certain quality measures, both objective and subjective. The perceptual disparity is a parameter that models the subjective perception of the user based on the disparity of the scene. Based on this model, we have other projects that give us a prediction of the image quality based on such perceptual disparity. These quality predictive models have been tested in a limited amount of data, but always obtaining positive results. Our objective in this TFG is to increase the amount of training data of this model, obtaining more quantity and more variety of data to see how the model generalizes to a broader and bigger number of training data. First, we increased the data in space and in time and we visualized the variation on PSNR and perceptual disparity of each. Then, we proceeded to analyze our new training data in a similar way as to the one performed previously. Lastly, we compared the new results to those of the previous paper, observing similar results and concluding that the model is generic for a bigger and broader amount of data.

Keywords

3D video, Free Viewpoint Video, Super Multiview Video, multiview, Peak Signal to Noise Ratio, PSNR, Multiview Perceptual Disparity Model, MVPDM.

Agradecimientos

En primer lugar, agradecer a mi tutor, Pablo, por todo el apoyo y el tiempo que me ha dedicado durante el desarrollo de este trabajo.

Agradecer también a todos los profesores que me han ayudado durante la carrera, en especial a Álvaro y Marcos, que me ayudaron cuando más lo necesitaba.

A mi familia, y a todos los amigos que he hecho durante estos cuatro años, en especial a Cosín, Juanito, Pol y Mario.

INDICE DE CONTENIDOS

1 Introducción 1
1.1 Motivación 1
1.2 Objetivos 2
1.3 Organización de la memoria 3
2 Estado del arte 5
2.1 3D video y Free Viewpoint Video 5
2.1.1 Super Multiview Video (SMV) 6
2.2 Introducción al problema de predicción de calidad de vistas. 7
2.2.1 Síntesis de vistas virtuales a partir de otras reales 7
2.2.2 Medida de calidad utilizadas: PSNR 8
2.2.3 Introducción al modelo previo basado en MVPDM (Multiview Perceptual Disparity Model). 10
3 Diseño y desarrollo 15
3.1 Introducción 15
3.2 Resultados del modelo analizado en [1] 15
3.3 Ampliación de los datos de análisis 19
3.3.1 Aumento de los escenarios y vistas virtuales 19
3.3.2 Modificación en la extracción de valores de PSNR 20
3.3.3 Modificación en el cálculo de $d_{Perc}near$ y $d_{Perc}Far$ 21
3.4 Optimización aplicada a las nuevas condiciones 21
4 Integración, pruebas y resultados 23
4.1 Introducción 23
4.2 Contenido utilizado 23
4.3 Ampliación de los datos: Análisis espacial 25
4.4 Ampliación de los datos: Análisis temporal 27
4.5 Resultados del análisis para los nuevos datos 29
4.5.1 Resultados para el análisis espacial 29
4.5.2 Resultado para la disparidad en Znear 33
4.5.3 Análisis para la variación temporal 34
4.6 Herramienta automática de generación de datos 37
5 Conclusiones y trabajo futuro 39
Referencias 41

INDICE DE FIGURAS

FIGURA 1: *DISPOSICIÓN DE LAS CÁMARAS EN UN SISTEMA FVV RESPECTO DE LA ESCENA* [4]..... 5

FIGURA 2: *DIAGRAMA DE BLOQUES DE UN SISTEMA SMV* [3]..... 6

FIGURA 3: *EJEMPLO DE INFORMACIÓN DE COLOR Y PROFUNDIDAD DE LA ESCENA.* 7

FIGURA 4*: EJEMPLO DE SÍNTESIS DE LA MISMA VISTA A PARTIR DE DOS PAREJAS DE CÁMARAS.* 8

FIGURA 5: *DISTINTAS DISPOSICIONES DE LAS CÁMARAS DE SMV [6]*..... 10

FIGURA 6: *VALORES DE DISPARIDAD EN FUNCIÓN DE LA DISTANCIA DE LOS OBJETOS* [6]. 11

FIGURA 7: *REPRESENTACIÓN DEL CÁLCULO DE* D_{PERC} *PARA LAS DOS FUNCIONES DE PESO* 12

FIGURA 8: *DISPARIDAD PERCEPTUAL DE UN ESCENARIO*..... 12

FIGURA 9: $D_{NEAR\ Y}\ \Theta_{ANG}$ *PARA UN ARRAY DE CÁMARAS NO LINEAL* 13

FIGURA 10: *RESULTADOS DE PSNR EN FUNCIÓN DE* D_{PERC}*NEAR Y* D_{PERC}*FAR* [1]..... 16

FIGURA 11: *APROXIMACIÓN MEDIANTE UN PLANO DE LOS VALORES ÓPTIMOS DE PSNR* [1] 17

FIGURA 12: *VALORES DE ERROR MÍNIMO MEDIO PARA DISTINTOS VALORES DE A Y B (MARCADO EN ROJO)* 17

FIGURA 13: *REPRESENTACIÓN DEL VALOR DE PSNR RESPECTO DE LA DISPARIDAD EN ZNEAR* [1].... 18

FIGURA 14: *ESCENARIOS UTILIZADOS PARA EL ANÁLISIS DE LAS SECUENCIAS.* 19

FIGURA 15: *COMBINACIONES POSIBLES DE SÍNTESIS DE VISTAS VIRTUALES PARA UN ARRAY DE 5 CÁMARAS*..... 20

FIGURA 16: *VISTAS DE LA CÁMARA CENTRAL PARA CADA UNA DE LAS SECUENCIAS.* 24

FIGURA 17**:** *REPRESENTACIÓN GRÁFICA DE LOS VALORES DE DISPARIDAD EN*..... 25

FIGURA 18*: REPRESENTACIÓN GRÁFICA DE LOS VALORES DE PSNR EN FUNCIÓN DE CADA CUADRO PARA LAS CUATRO SECUENCIAS UTILIZADAS.* 26

FIGURA 19*: REPRESENTACIÓN GRÁFICA DE LOS VALORES DE DISPARIDAD PERCEPTUAL EN FUNCIÓN DEL ÍNDICE DE CÁMARA PARA LAS CUATRO SECUENCIAS UTILIZADAS.* 27

FIGURA 20: *EJEMPLO DE ANÁLISIS ESPACIAL PARA UN ARRAY DE 9 CÁMARAS Y 6 CUADROS POR SECUENCIA* 29

FIGURA 21*: APROXIMACIÓN ÓPTIMA DEL PSNR A PARTIR DE LOS VALORES DE* D_{PERC}*NEAR Y* D_{PERC}*FAR PARA LA FUNCIÓN POLINÓMICA.* 30

FIGURA 22*: APROXIMACIÓN ÓPTIMA DEL PSNR A PARTIR DE LOS VALORES DE $D_{PERC}NEAR$ Y $D_{PERC}FAR$ PARA LA FUNCIÓN SIGMOIDE.* 31

FIGURA 23: *RESULTADOS DEL ERROR MÍNIMO MEDIO PARA UN RANGO DE VALORES DE A Y B (VALOR ÓPTIMO SEÑALADO EN ROJO)* 31

FIGURA 24*: APROXIMACIÓN ÓPTIMA DE PSNR A PARTIR DE LOS VALORES DE DISPARIDAD MÁXIMA DE LA ESCENA.* 33

FIGURA 25: *EJEMPLO DE ANÁLISIS TEMPORAL PARA UN ARRAY DE 9 CÁMARAS Y 6 CUADROS POR SECUENCIA.* 34

FIGURA 26: *REPRESENTACIÓN DE LOS VALORES DE PSNR EN FUNCIÓN DE $D_{PERC}NEAR$ Y $D_{PERC}FAR$* 35

FIGURA 27: *REPRESENTACIÓN DE LOS VALORES DE PSNR EN FUNCIÓN DE $D_{PERC}NEAR$ Y $D_{PERC}FAR$* 35

INDICE DE TABLAS

TABLA 1: VALORES DE A Y B PARA LOS QUE LA APROXIMACIÓN ES ÓPTIMA. 18

TABLA 2*: CARACTERÍSTICAS DE LAS SECUENCIAS DE VÍDEO SMV UTILIZADAS.* 24

TABLA 3: VALORES ÓPTIMOS DE A Y B PARA LA FUNCIÓN POLINÓMICA Y SIGMOIDE. 31

TABLA 4: COMPARATIVA VALORES DE ERROR PARA D_{PERC} Y $Z_{NEAR.}$ 33

TABLA 5: VALORES DE ERROR PARA LAS DOS FUNCIONES DE PESO. 35

1 Introducción

1.1 Motivación

Las tecnologías de visualización de vídeo en 3D han crecido considerablemente en los últimos años, siendo cada vez más los avances tecnológicos en esta área, así como en el desarrollo de sistemas audiovisuales más económicos y con menos limitaciones.

Uno de los tipos de contenido que aporta soluciones a la problemática de los sistemas de reproducción de vídeo 3D del cansancio o fatiga visual, además de a la necesidad de utilizar gafas para la visualización del contenido, es el vídeo súper multivista (SMV). Este tipo de tecnología permite visualizar diferentes perspectivas de la escena según el movimiento de nuestra cabeza, o la elección del usuario, gracias a los arrays de numerosas cámaras, reales o virtuales, que generan el contenido visualizado.

Sin embargo, este novedoso sistema de visualización de vídeo 3D presenta algunos problemas relacionados con el coste de la producción del mismo. Son necesarias cantidades muy grandes de cámaras y sistemas que transmitan los datos capturados, lo que supone una incapacidad técnica de transmisión de tal cantidad de datos para las redes de transmisión actuales [15]. Además, son necesarias cantidades muy grandes de cámaras y sistemas que transmitan los datos capturados, lo que supone una incapacidad técnica de transmisión de tal cantidad de datos para las redes de transmisión actuales. Por otro lado, se propone como la solución propuesta de generar vistas virtuales para optimizar gastos. En estas vistas se producirá una pérdida de calidad, que a veces es impermisible a la hora de visualizar los contenidos.

Para generar unas vistas virtuales que nos permitan sustituir estas por las reales, sería útil tener una herramienta que permita definir la densidad de cámaras reales necesarias para que la calidad de las vistas virtuales esté por encima de un nivel deseado.

Como solución, se proponen modelos que, entrenados mediante determinadas medidas de calidad, y relacionando dichas métricas con datos que aporta la escena, permitan predecir la calidad de las vistas generadas de forma objetiva, pero de una forma aproximada a la visualización subjetiva del espectador. Uno de estos sistemas, el Multiview Perceptual Disparity Model (MVPDM) permite realizar este modelado a partir de la información sobre la colocación de las cámaras y de la profundidad de la escena, y será a partir del cual trabajaremos.

En esta línea, ya tenemos algún trabajo basado en el MVPDM, y será a partir del cual partiremos, para analizar su capacidad de generalización.

1.2 Objetivos

El principal objetivo de este Trabajo de Fin de Grado es profundizar en el modelo preliminar de predicción de la calidad de vistas virtuales para vídeo multivista, presentado en [1].

En las pruebas realizadas en el modelo en el trabajo anterior, los datos de entrenamiento utilizados no permitían analizar toda la cantidad y variedad de datos posibles que ofrecían las secuencias analizadas, tanto en el tiempo como en el espacio.

Este modelo sirvió de precedente para ver que podía funcionar correctamente, sin embargo, a la hora de hacer un uso práctico del mismo, será necesario haber probado el modelo con toda la variabilidad y cantidad de datos posible, así como recalcular los parámetros óptimos utilizados en la prueba inicial del modelo, pues con la nueva cantidad de datos es posible que estos hayan variado.

De esta forma, se buscará evaluar su capacidad de generalización a diferentes configuraciones de cámaras y contenido, así como aumentar y mejorar la cantidad de datos de entrenamiento, solventando así algunas de sus limitaciones.

Además, se buscará proporcionar herramientas que permitan facilitar la automatización del proceso de generación de datos para el entrenamiento del modelo. De esta forma, los datos de entrenamiento serán totalmente parametrizables, ahorrando tiempo para cada prueba con un conjunto de datos distinto.

1.3 Organización de la memoria

Para organizar la memoria hemos seguido la siguiente estructura:

- **Capítulo 2: Estado del arte.**

  En este capítulo hablaremos de los contenidos básicos que son necesarios conocer para poder entender el trabajo fácilmente. En primer lugar, describiremos el contenido sobre el que se basa el trabajo, el video 3D Video y FVV, así como el SMV. Después, entraremos de lleno en las bases que utilizan los trabajos anteriores al nuestro que buscan estudiar el mismo problema. Se verá cómo se realiza la síntesis de vistas, las medidas de calidad utilizadas (PSNR) y una introducción al modelo sobre el que se parte para realizar el trabajo, el MVPDM.

- **Capítulo 3: Diseño y desarrollo.**

  En este capítulo nos centraremos en explicar las limitaciones del trabajo previo de [1]. Después, hablaremos de las mejoras que implementaremos en los datos, como son la ampliación en la información de las cámaras y en el cálculo de medidas de PSNR, así como el cambio en el uso de la información de profundidad en el cálculo de algunos parámetros.

- **Capítulo 4: Integración, pruebas y resultados.**

  En este capítulo hablaremos sobre los resultados que tienen que ver con las mejoras explicadas en el capítulo 3. En primer lugar, se describirá el contenido que utilizaremos para las pruebas del modelo. Después se mostrarán los resultados sobre la variación espacial y temporal de los datos, y los resultados finales para esos datos de entrenamiento. Finalmente, se describirá la herramienta creada para la generación de los datos utilizados.

- **Capítulo 5: Conclusiones y trabajo futuro.**

  Para finalizar, se reflexiona sobre aquellas conclusiones que nos deja el trabajo, y se plantean las posibles líneas de trabajo futuro que podría haber para mejorarlo.

2 Estado del arte

En este estado del arte se expondrán todos los conocimientos previos necesarios para la comprensión total de nuestro trabajo. En primer lugar, definiremos las bases explicando brevemente los sistemas de vídeo 3D y Free Viewpoint Video, para después entrar en el sistema de vídeo sobre el que aplicaremos el modelo de calidad de vistas virtuales, el Super Multiview Video (SMV). Después, continuaremos explicando el proceso de generación de vistas virtuales que nos permitirá ahorrar en la transmisión del SMV. Estas vistas virtuales tienen errores que empeoran la percepción, por lo que necesitaremos un método que nos permita medir su calidad. Hablaremos de algunas de las medidas de calidad objetiva utilizadas en el SMV, y en concreto, del PSNR, que será la que utilizaremos. Por último, hablaremos sobre un modelo previo que permite predecir estas medidas de calidad, basado en otro modelo que permite parametrizar las características de la escena que influyen en la percepción del usuario, el Multiview Perceptual Disparity Model (MVPDM).

2.1 3D video y Free Viewpoint Video

Los sistemas de vídeo 3D son uno de los principales campos de investigación en el desarrollo de sistemas de reproducción de contenido multimedia. Éstos permiten al usuario percibir el contenido con una sensación de profundidad real de la escena reproducida, gracias a la estereoscopía, encargada de capturar la información de manera que al reproducirla tengamos una imagen ligeramente diferenciada en cada ojo. Este proceso es análogo a la diferencia de perspectivas percibida por el ojo humano en una situación de visualización de una escena real, de manera que el cerebro junta estas dos imágenes generando una imagen tridimensional. Estos sistemas están asentados en la actualidad en el mercado, teniendo disponibilidad para usarlos en diferentes plataformas, como pueden ser televisiones, videoconsolas, cines, dispositivos móviles, etc. [2].

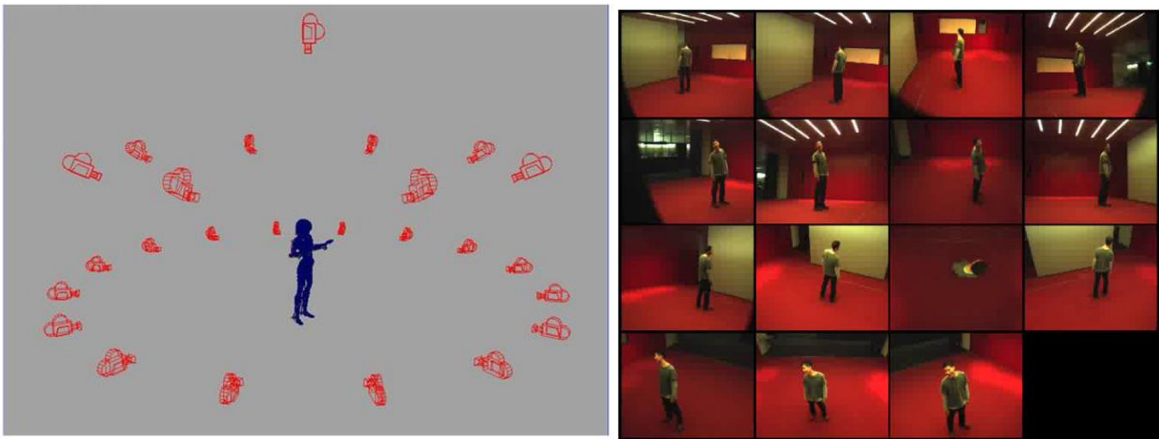


Figura 1: *Disposición de las cámaras en un sistema FVV respecto de la escena* [4].

Los sistemas de vídeo Free Viewpoint Video (FVV), son compatibles con las tecnologías 2D y 3D, y en ellos el usuario tiene la capacidad de elegir desde qué punto de vista ver la escena. Para poder generar este tipo de contenido, es necesaria la utilización de una cantidad considerable de cámaras que graben la escena desde los distintos puntos de vista. Gracias a esta información, es posible generar vistas virtuales de la escena a partir de otras reales, de manera que se pueda sintetizar cualquier punto de vista arbitrario. Esta plataforma de información espacial está asentada en campos como la reproducción de videojuegos, eventos deportivos, aplicaciones culturales, de videovigilancia, etc. [3].

2.1.1 Super Multiview Video (SMV)

A parte del contenido de vídeo 3D y FVV, uno de las tipos de visualización que más ha crecido en los últimos años, y que ofrece una experiencia 3D sin gafas de calidad, es el video SMV (Super Multiview Video). Éste es un sistema de video autoestereoscópico cuya densidad de vistas sobre la escena es muy alta, lo que lleva a una calidad superior que la de los sistemas de video 3D y FVV tradicionales [6].

Los sistemas de visualización 3D han crecido mucho durante los últimos años gracias a la llegada de las pantallas 3D al mercado. Sin embargo, este tipo de contenido no se ha asentado entre los consumidores, debido en gran parte a dos limitaciones que tendrán mucho que ver en el análisis que llevaremos a cabo en nuestro trabajo. En primer lugar, debido a la poca cantidad disponible de video 3D de alta calidad. Esta falta de calidad, se puede deber a la poca cantidad de información desde distintos puntos de vista de la escena con la que se genera el video 3D, lo que puede provocar fatiga visual y lleva a una experiencia del usuario negativa. Por otra parte, la necesidad de llevar puestas unas gafas para disfrutar del contenido no ayuda, por ello se está trabajando muy activamente en el desarrollo de sistemas 3D sin gafas, aunque aún no ofrecen una calidad muy alta para el precio que tienen [6].

Los sistemas SMV cumplen el requisito también llamado *SMV*, por el cual la distancia entre dos vistas adyacentes debe ser menor que el diámetro de la pupila. Esto alivia la fatiga visual producida por la dificultad en el enfoque que presentan las pantallas estereoscópicas. Esta dificultad es conocida como conflicto acomodación-convergencia, y se debe a que el punto en el que convergen los ojos al percibir la escena, es diferente al punto de enfoque, lo que hace que el ojo esté enfocando constantemente, llevando a la fatiga visual [12]. El SMV, además, tiene menos limitaciones que las pantallas holográficas, ya que tiene más alternativas a la hora de proyectar estas vistas al usuario, entre las que destacan la matriz de luz enfocada, la proyección múltiple, o la multiplexación en el tiempo [6].

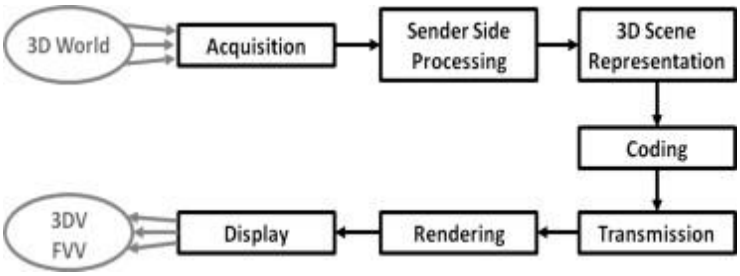


Figura 2: *Diagrama de bloques de un sistema SMV* [3].

Para generar el contenido de video SMV, y como se puede ver en la Figura 2, es necesaria una serie de fases por las que pasa la información. Al tener una cantidad de vistas y datos tan grande, como método de ahorro de información, se propone la compresión de las vistas del SMV durante la codificación. Después de la codificación, se transmitirán las vistas, que se podrán visualizar correctamente una vez se procesen en el decodificador.

Estos sistemas, a pesar de comprimir la información a transmitir, necesitan de un coste tanto económico (cantidad muy grande de cámaras y equipo de procesamiento y transmisión del video capturado) como computacional muy grande. Como solución se propone la síntesis de algunas de las vistas a partir de la información de la escena de otras vistas reales, de manera que se pueda tener toda la información de la escena sin necesidad de una cantidad tan grande de cámaras. La forma en que se produce la síntesis de video la analizaremos más adelante.

2.2 Introducción al problema de predicción de calidad de vistas.

Como hemos explicado anteriormente, es necesario reducir el número de vistas transmitidas como solución al coste computacional tan grande que supone. Como solución, se propone la síntesis de vistas intermedias de la escena a partir de otras, de manera que se pueda reducir el número de vistas capturadas y transmitidas. Sin embargo, en estas vistas virtuales generadas se producen distorsiones que empeoran la percepción del video. Por tanto, necesitaremos un modelo a partir del cual podamos predecir la calidad del video a partir de información de la configuración de las cámaras y de la escena. De esta forma, se podrá definir una densidad de cámaras mínima necesaria, para la cual la calidad de la percepción del usuario sea buena.

2.2.1 Síntesis de vistas virtuales a partir de otras reales

Para la síntesis de las vistas virtuales que se propone, son necesarias dos tipos de información diferentes. En primer lugar, se necesitará la información de color de la escena grabada, y por otro lado necesitaremos la información de profundidad. Esta información se muestra como un rango de valores de gris, que indican a cuánta distancia se encuentra la escena de la cámara. La profundidad es una de las posibles representaciones de la geometría de la escena. De esta forma, cuanto mayor sea el nivel de gris, más cerca se encontrará el objeto de la cámara, tal y como se muestra en la Figura 3. El formato en el que nos hemos basado, y que incluye la información de color y de profundidad por cámara, se llama *Multiview + Depth* (MVD), y está contenido en el estándar 3D-HEVC [4].

Figura 3: *Ejemplo de información de color y profundidad de la escena.*

Esta información tiene sus limitaciones, ya que en la síntesis de las vistas virtuales se producirán artefactos que afectarán negativamente a la calidad, en parte, debido a que los de profundidad no están estimados de manera precisa. En el caso de las vistas virtuales que no son generadas a partir de cámaras reales, los mapas de profundidad no tendrán la información exacta, como ocurre en las vistas generadas a partir de ordenador, sino que será una estimación de los mismos, por lo que puede dar lugar a errores.

Además, también puede influir negativamente la distancia entre las vistas virtuales sintetizadas y las cámaras de referencia, a partir de las cuales se han generado. Siendo a mayor distancia cuanto más disminuya la calidad, como se puede apreciar en la Figura 4. El proceso de síntesis se produce mediante algoritmos ***Depth Image Based Rendering (DIBR)*** [12]. En ellos, se sintetizan las vistas virtuales entre las cámaras a partir del desplazamiento de los píxeles y objetos de una vista real, en función de su disparidad. Para la síntesis de las vistas el programa utilizado es el VSRS 4.0, de MPEG [5].

Figura 4*: Ejemplo de síntesis de la misma vista a partir de dos parejas de cámaras.*

En la Figura 4 podemos observar dos vistas virtuales generadas a partir de otras reales. La primera de ellas, está generada utilizando dos vistas reales situadas a una cámara de distancia. En ella, podemos observar que la percepción subjetiva es muy buena ya que no hay ningún error o artefacto en la imagen que empeore la visualización. La segunda vista, para un array de 80 cámaras, se encuentra a diez cámaras de distancia de las vistas reales a partir de las cuales se ha sintetizado. En ella, podemos observar que entre la ardilla y la mariposa hay un artefacto que, aunque no lleva a tener una muy mala visualización, empeora la percepción del video.

2.2.2 Medida de calidad utilizadas: PSNR

Como solución al problema de los errores generados en las vistas sintetizadas, como se muestra en la Figura 4, necesitaremos una medida de calidad que nos permita ver cómo de buena es la percepción del usuario. Aunque lo ideal sería realizar el análisis para medidas subjetivas, éstas pruebas son mucho más costosas económicamente y requieren de un gasto de tiempo mucho mayor, por lo que utilizaremos una de las medidas objetivas que nombraremos a continuación [11].
Dentro de las medidas objetivas de calidad de video que se utilizan para medir la calidad del SMV, y que nos permitirán medir la calidad de las vistas virtuales que generaremos, tenemos varias posibilidades, entre las que se encuentran PSNR, SSIM, VQM, y 3DSwlM. Tras ver los resultados de cada una de ellas en los resultados para las pruebas del trabajo

mostrado en [1], y dado que lo que se busca es una relación con la percepción del usuario, se obtienen las siguientes conclusiones.

- PSNR: El PSNR (Peak Noise to Signal Ratio) es una de las medidas más extendidas en la actualidad. Está basada en el cálculo del error cuadrático medio (MSE) de cada imagen pixel a pixel, sin embargo tiene un coste computacional bajo [8].

- SSIM: Está basada en la medida de la diferencia de estructura, luminancia y contraste entre dos imágenes [9].

- VQM: Es una medida basada en la variación de características que dependen de la percepción del usuario, consiguiendo así medir variaciones en el tiempo, espacio y crominancia entre un video real y uno virtual [10].

- 3DSwlM: Está basado en que los cambios en los elementos de la escena que sean considerados como humanos tengan más peso que el resto de elementos [10].

Para el caso del PSNR, en su prueba con las cuatro secuencias analizadas la calidad de las vistas virtuales disminuye a medida que aumenta la distancia entre éstas y las vistas reales o de referencia. Esto es algo a tener en cuenta pues esa distancia también será determinante en la parametrización que mostraremos en el siguiente punto [1].

En cuanto al SSIM, los resultados que ofrece en los trabajos anteriores propuestos son similares a los de PSNR, sin embargo, se basa en la estructura de la imagen por lo que cualquier error en el cálculo de la información de profundidad explicado anteriormente podría afectar muy negativamente en las medidas [1].

Para el VQM, los resultados obtenidos fueron similares a los de PSNR, y para el 3DSwlM, los resultados no fueron coherentes con los anteriores.

Finalmente, se eligió hacer el análisis para medidas de PSNR, ya que dentro de las medidas objetivas que reflejan la calidad de las vistas virtuales de forma correcta es la más utilizada en el campo de análisis de medidas de calidad de video. Además, tras ver los resultados finales del análisis del trabajo anterior, se vio que era una buena estimación a falta de entrenar el modelo con más datos, por lo que continuaremos utilizando esta medida en nuestro trabajo [1].

2.2.3 Introducción al modelo previo basado en MVPDM (Multiview Perceptual Disparity Model).

Para poder predecir las medidas de calidad nombradas en el capítulo anterior, será necesario tener un modelo que nos permita hacerlo a partir de los valores de determinados parámetros de la escena. Uno de los modelos previos que se encargan de realizar esta predicción es el mostrado en [1]. Este modelo, utiliza la base del modelo MVPDM [6], la disparidad perceptual entre cámaras ($\boldsymbol{d_{perc}}$), para realizar una aproximación de los valores de PSNR en función de los de $\boldsymbol{d_{perc.}}$

El MVPDM es un modelo que se encarga de aproximar una relación entre valores de calidad y determinados parámetros de la escena, para medir la percepción subjetiva del usuario sobre la visualización. En la práctica, se ha demostrado una alta correlación entre los resultados de percepción subjetiva y el modelo. Además, el modelo aporta información sobre algunos parámetros que serán utilizados para garantizar una experiencia positiva en la visualización de video SMV [6].

El MVPDM está basado en la disparidad perceptual entre dos cámaras, que depende de cinco factores principales: el campo de visión (distancia focal), el grado de rotación de las cámaras, el ángulo de la cámara, la profundidad de la escena y la convergencia entre las cámaras. Sin embargo, este parámetro es independiente de la configuración de cámaras utilizada. Por tanto, se analizarán tres tipos de escenarios diferentes en función de la disposición del array de cámaras: de forma lineal, de forma lineal convergente, o en forma de arco, como se puede ver en Figura 6.

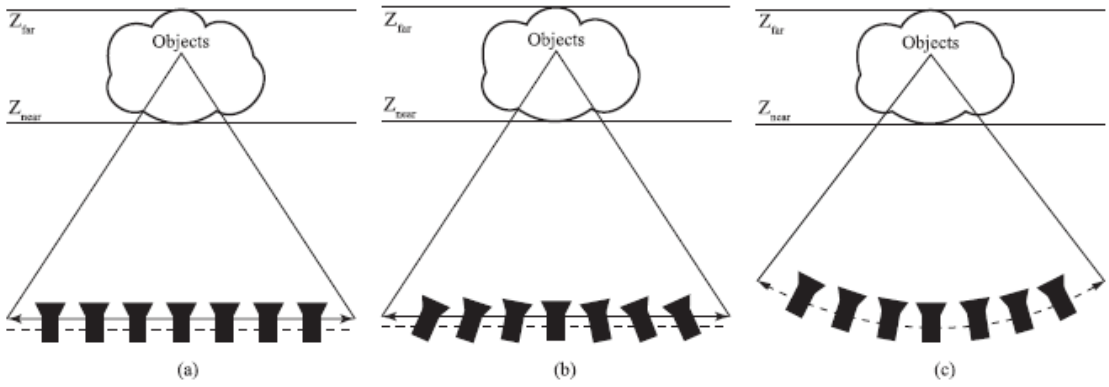


Figura 5**:** *Distintas disposiciones de las cámaras de SMV* [6].

El MVPDM propone una parametrización de tres elementos que se ha visto que tienen gran influencia en la percepción del usuario. Estos elementos son el rango de visión, la densidad de las vistas, y la velocidad con la que se cambiará de punto de vista. El parámetro que busca relacionar estos factores con la percepción subjetiva es la disparidad. Sin embargo, este parámetro depende de la profundidad de la escena, ya que los factores anteriores dependen de ella, y para parametrizar la secuencia, sería más útil tener valores representativos de cada uno de ellos, y no tenerlos como función de otros parámetros.
La disparidad, como vemos en la Figura 6, es mayor de cero cuando los elementos de la escena se encuentran lejos de la cámara, y menor de cero a medida que estos se acercan a la cámara.

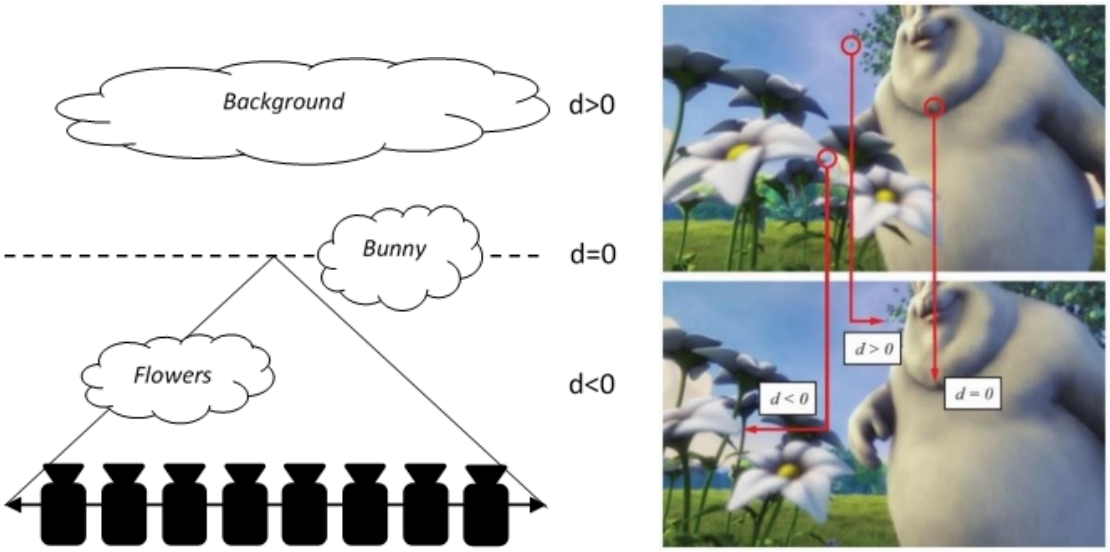


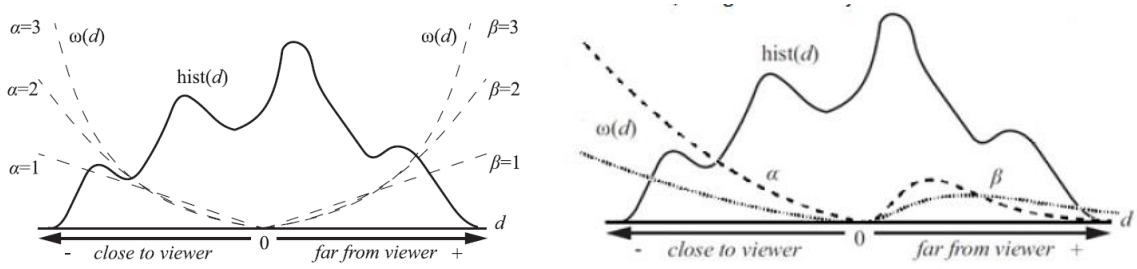
Figura 6: *Valores de disparidad en función de la distancia de los objetos* [6]**.**

Para solucionar este problema en la parametrización, se utiliza la disparidad, teniendo en cuenta la profundad del objeto más cercano y más lejano de la escena. A partir de estos factores, el modelo define un nuevo parámetro, la disparidad perceptual $\boldsymbol{d_{perc}}$, que representa la distribución de la disparidad entre las vistas de la escena. Es el elemento básico del MVPDM [6].

$$\boldsymbol{d_{perc} = \int_{d_{min}}^{d_{max}} hist(\delta)\, \omega(\delta) d\delta} \qquad \textbf{(1)}$$

La variación de $\boldsymbol{d_{perc}}$ en función de la distancia focal a los objetos de la escena, está determinada por una función de peso $\boldsymbol{\omega}$ **(d)** y de un histograma de disparidades de la escena ***hist (d)*** [6]**.** La función de peso depende de dos parámetros que determinarán la importancia de los objetos de la escena en función de lo lejos o cerca que se encuentren. Un parámetro α, cuyo valor en la función dependerá del peso que tienen los objetos que se encuentran más cercanos a la cámara en la escena, y un parámetro β que se encargará de establecer la ponderación de los objetos más alejados de la cámara. Para el trabajo desarrollado en [1], se utilizaron dos funciones de peso distintas, una función polinómica y una función sigmoide, con las siguientes expresiones.

$$\boldsymbol{\omega(d) = \begin{cases} |d|^{\alpha} & d \leq 0 \\ |d|^{\beta} & d \leq 0 \end{cases}} \quad \textbf{(2)} \qquad \boldsymbol{\omega(d) = |d|^{\alpha} \frac{1}{1 - e^{\beta d}}} \quad \textbf{(3)}$$

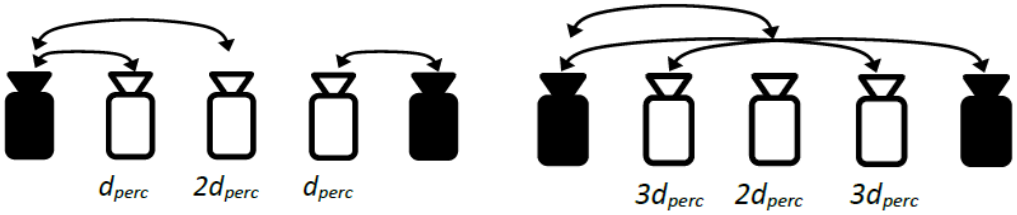


Función de peso polinómica Función de peso sigmoide

Figura 7: *Representación del cálculo de d_{perc} para las dos funciones de peso*

Como podemos apreciar en la Figura 7, las dos funciones se comportan igual para los valores de disparidad negativos, es decir, para los objetos que se encuentran por detrás del plano de convergencia, los más cercanos al observador. Sin embargo, tienen un comportamiento distinto para los objetos situados más lejos en la escena. En la función sigmoide, solo se tienen en cuenta los objetos alejados de la escena cercanos al plano de convergencia, sin dar ninguna importancia a aquellos que se encuentran a mucha distancia de la misma.

En ambas funciones, para calcular la disparidad perceptual entre una vista de referencia y una virtual sintetizada, se utiliza el mismo procedimiento. El primer paso consiste en calcular la disparidad perceptual entre la cámara central de referencia y su consecutiva. Después, se multiplicará el valor calculado de $\boldsymbol{d_{Perc}}$ por el número de cámaras de distancia a la que esté la cámara virtual generada, obteniendo así la $\boldsymbol{d_{Perc}}$ entre la cámara real y la virtual. La $\boldsymbol{d_{Perc}}$ a la cámara más cercana será la $\boldsymbol{d_{Perc}near,}$ y aquella a la cámara más lejana $\boldsymbol{d_{Perc}far.}$ Este método de cálculo se modificará en nuestra aproximación del modelo como se explicará más adelante [1].



a) Disparidad perceptual a la cámara cercana *a) Disparidad perceptual a la cámara lejana*

Figura 8: *Disparidad perceptual de un escenario*

Por último, se realiza una normalización frente a la resolución horizontal de la $\boldsymbol{d_{Perc}}$, y así poder realizar el análisis para secuencias de diferente resolución, como se muestra en la siguiente expresión.

$$\boldsymbol{d_{horizontal}}(\%) = \frac{\mathbf{100}}{\boldsymbol{ancho}}\ \boldsymbol{d} \qquad \textbf{(4)}$$

En el modelo de calidad de vistas previo [1], se estudia la aproximación de los valores de PSNR a partir de los valores de $\boldsymbol{d_{Perc}near}$ y $\boldsymbol{d_{Perc}far}$, como hemos explicado anteriormente. Se busca un modelo a partir del cual, mediante valores de calidad objetivos (PSNR), tengamos una medida de calidad fiable de la percepción subjetiva del usuario.

Además de utilizar la base del modelo MVPDM explicado, también se realizó una aproximación del modelo para dos parametrizaciones alternativas. Estos parámetros son más simples ya que están basados en alguno de los elementos de la $\boldsymbol{d_{Perc}}$, por lo que dan una aproximación de la calidad de forma más precisa. Por el contrario, su correlación con la percepción del usuario es menor que la de la disparidad perceptual [6]. El primer parámetro que se utilizó, fue la disparidad máxima de la escena ($\boldsymbol{d_{near}}$). Ésta se calculó como la disparidad entre las dos cámaras del centro del array en el punto en el que un objeto de la escena se encuentra más cerca del array de cámaras. El segundo parámetro utilizado, fue el ángulo entre las dos cámaras centrales, $\boldsymbol{\theta_{ang}}$. Se calculó el ángulo entre las distancias al objeto más cercano, en caso de que el array fuese lineal, y entre las distancias al plano de convergencia, en el caso de que el array estuviese en forma curva.

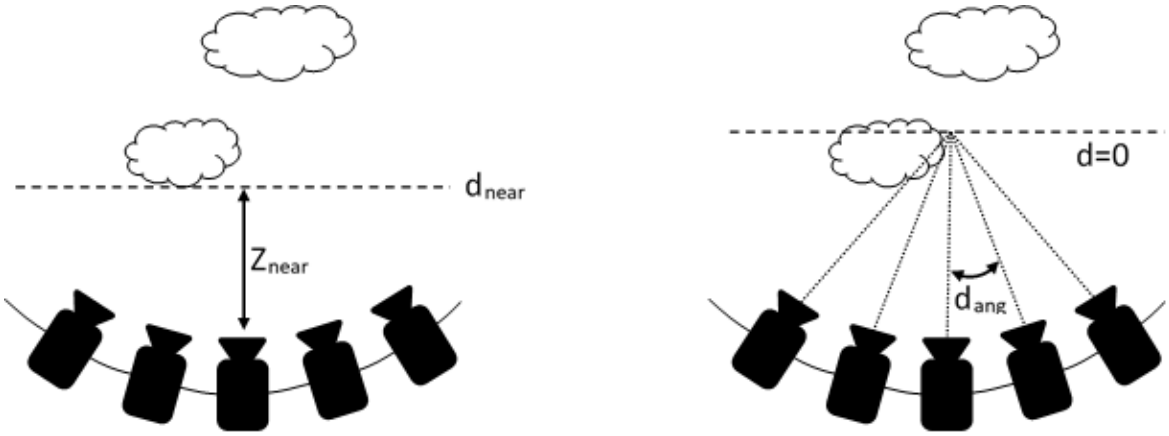


Figura 9: *d_{near} y θ_{ang} para un array de cámaras no lineal*

En la Figura 9, podemos ver un ejemplo de la cámara respecto de la que se calculó la disparidad (disparidad en $\boldsymbol{Z_{near}}$) y de $\boldsymbol{\theta_{ang}}$.

Los resultados de este modelo previo se verán en el siguiente capítulo, pues servirá de precedente para las mejoras que aportaremos, basadas en la ampliación y mejora de los datos de prueba.

3 Diseño y desarrollo

3.1 Introducción

En este capítulo haremos un repaso del análisis de los resultados del modelo de calidad de vistas mostrado en [1], a partir del cual evaluaremos su capacidad de generalización a una cantidad y variedad mayor de datos. En este modelo se busca una parametrización que permita una estimación precisa de la calidad subjetiva del video. Esta parametrización está basada en el MVPDM explicado en el Capítulo 2.2.3, y, por tanto, en la disparidad entre cámaras. Después de ver los resultados previos, mostraremos las variaciones en los datos de entrenamiento, que utilizaremos para probar de nuevo el modelo. En primer lugar, hablaremos acerca del aumento de los datos en el espacio (aumento de configuraciones de las cámaras para un array determinado) y en el tiempo (desagregación de valores de PSNR). Por último, hablaremos sobre la modificación llevada a cabo en el uso de los datos de profundidad de la escena, utilizando la de cada cámara de análisis. Asimismo, hablaremos de la optimización del modelo tras entrenarlo con los nuevos datos.

Los resultados de la nueva aproximación se compararán y analizarán con los del modelo anterior en el Capítulo 4, donde también se seguirán los pasos análogos del modelo previo para los nuevos datos de prueba. En el Capítulo 4, además, se mostrará cómo se han desarrollado las nuevas herramientas automáticas de generación de datos. Éstas, facilitarán la realización de pruebas con una variedad de datos mayor. Por tanto, nos permitirán generar datos que sirvan para realizar un análisis de distinto tipo (temporal o espacial), mucho más rápida y eficientemente.

3.2 Resultados del modelo analizado en [1]

Tras ver los resultados del modelo en [1], que mostraremos a continuación, observamos que es una buena aproximación a lo buscado, pero que tiene algunas limitaciones.

1. Los datos son suficientes para aproximar las medidas de calidad, pero si queremos que este sea fiable para todo tipo de secuencias y variedad de situaciones, será necesaria la ampliación de los datos tanto en cantidad como en variedad.

2. La forma en que se agregan algunos datos es limitada y está hecha a partir de aproximaciones y no cálculos exactos. En cuanto al uso de los valores de PSNR, por ejemplo, utilizamos únicamente un valor para toda una secuencia, sin importar la variación temporal. En el caso del cálculo de disparidad perceptual entre cámaras, se toma siempre como profundidad de la escena la de la cámara central, y no la que ve cada una de las cámaras.

Por tanto, se han ampliado y variado los datos del análisis del que partimos, con el fin de ver cómo responde el modelo a estos nuevos datos de entrenamiento; es decir, para ver cómo generaliza.

En el trabajo desarrollado en [1] se realiza una aproximación al modelo basado en MVPDM en el cual se busca estudiar una relación entre los valores de PSNR en función de la disparidad perceptual. En primer lugar, se realiza el análisis viendo la relación de los valores de PSNR respecto de la disparidad perceptual entre la vista virtual generada y la cámara de referencia más cercana ($\boldsymbol{d_{Perc}Near}$). Tras visualizar los primeros resultados, la

relación observada no es lineal como se esperaba, por lo que se aplica el logaritmo en base diez a la disparidad a la cámara cercana. Esto facilita unos resultados que en primer lugar parecen guardar una relación lineal entre los valores de PSNR y los de disparidad perceptual, y también son coherentes en su medida del PSNR. La explicación es que éste disminuye a medida que se alejan las vistas virtuales de las de referencia, es decir, a medida que aumenta la disparidad perceptual. Para este caso, los resultados tanto de la función polinómica (2) como de la sigmoide (3) son similares. Pero dado que la hipótesis es que la calidad de la escena también depende de la disparidad perceptual a la cámara más alejada, se repitió el experimento pero cambiando la disparidad por aquella entre la cámara virtual y la real más alejada (***$d_{PercFar}$***). Finalmente, se estudió la relación entre los valores de PSNR, ***$d_{PercNear}$*** y ***$d_{PercFar}$***, obteniendo los siguientes resultados mostrados en la Figura 10, Figura 11 y Figura 13.

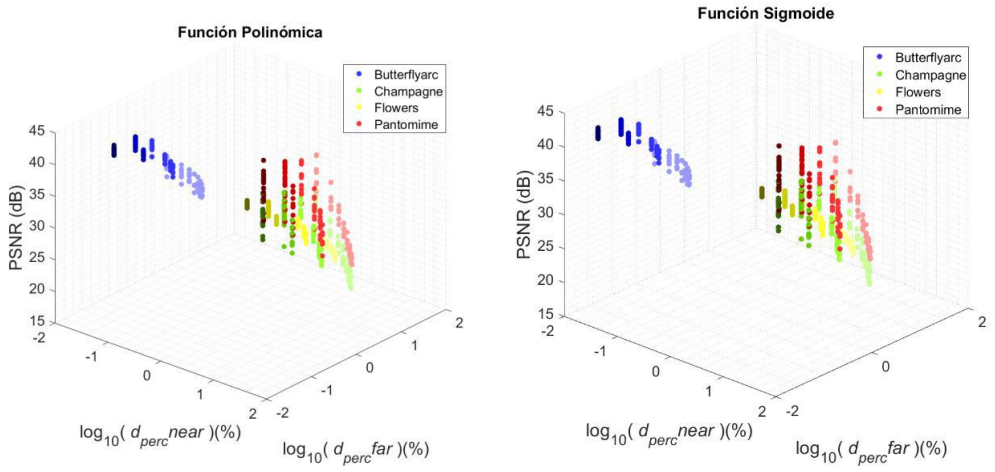


Figura 10: *Resultados de PSNR en función de $d_{Perc}Near$ y $d_{Perc}Far$* [1].

Dado que esta aproximación se realizó para unos valores de α y β fijos propuestos en el MVPDM [6], el siguiente paso fue buscar aquellos valores de dichos parámetros que ofrecieran la aproximación más parecida a la deseada. Se buscó que la relación entre los valores de PSNR Y ***d_{perc}***, mostrada en la Figura 10, fuese lo más parecida a un plano que nos permitiese aproximar los valores de PSNR a partir de los de disparidad perceptual, sin necesidad de calcularlos, como se puede apreciar en la Figura 11. Al realizar la optimización conjunta, se normalizaron los valores de ***$d_{PercNear}$*** y ***$d_{PercFar}$*** para los datos de las cuatro secuencias. Esta optimización también será necesaria de hacer una vez ampliemos los datos de entrenamiento, y la explicaremos en profundidad en el Capítulo 3.4. Los resultados finales obtenidos fueron los siguientes.

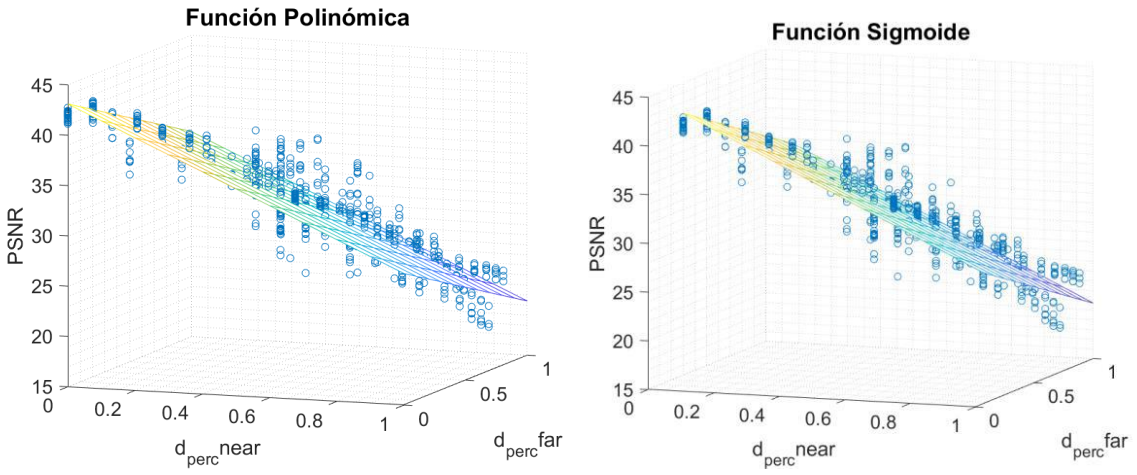


Figura 11: *Aproximación mediante un plano de los valores óptimos de PSNR* [1]

Como podemos observar en la Figura 11, el plano aproximado en los resultados finales es similar para la función polinómica que para la función sigmoide. En el caso de la función sigmoide, el hecho de dar más importancia a los objetos más cercanos influye únicamente en los valores óptimos de α y β que se utilizan en la función, pero el resultado es similar. Por tanto, ya que en los trabajos previos no hay una diferencia sustancial en los resultados para las dos funciones, se mantienen las posibilidades de utilizar dichas funciones en nuestro análisis para los nuevos datos de entrenamiento.

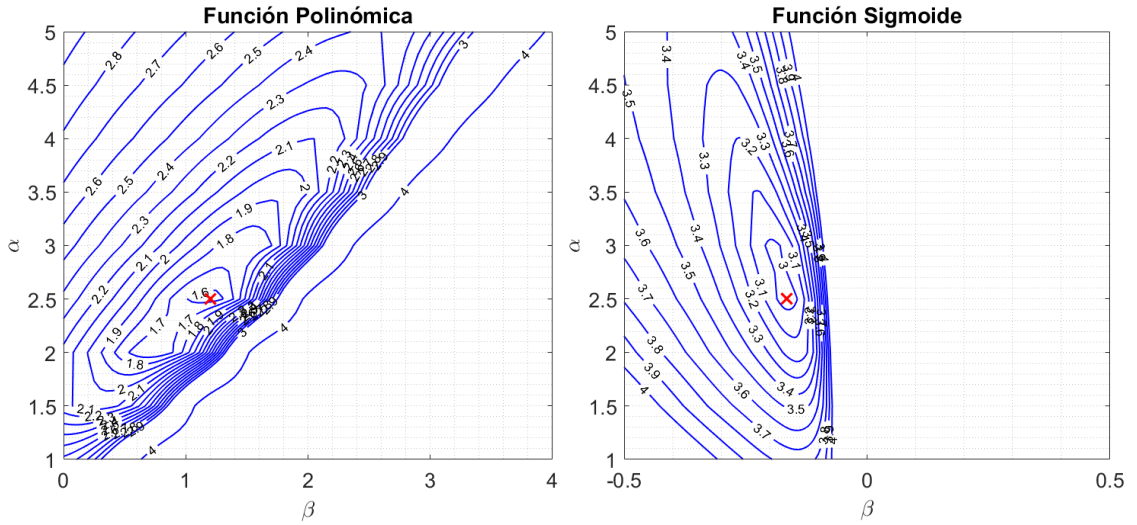


Figura 12: *Valores de error mínimo medio para distintos valores de α y β (marcado en rojo)*

Tabla 1: Valores de α y β para los que la aproximación es óptima.

	Función polinómica	Función sigmoide
α óptimo	2.50	2.50
β óptimo	1.25	-0.24
Error	1.52	1.50

Como podemos ver en la Tabla 1, los valores de α son iguales para las dos funciones, por lo que ambas darán la misma importancia a los objetos cercanos de la escena. En cuanto al valor de β, la función polinómica tiene un valor óptimo de 1.25, por lo que dará cierta importancia también a los objetos cercanos. No ocurrirá lo mismo con la función sigmoide, para la cual su valor óptimo de β es -0.24, por lo que tendrán mayor relevancia los valores de disparidad negativos y cercanos a cero.

Por último, se realizó una aproximación de los valores de PSNR basada en la máxima disparidad de la escena d_{near}. Ésta se calculó como la disparidad entre las dos cámaras centrales en el punto de la escena más cercano al array de cámaras, como se explica en el Capítulo 2.2.3. Este análisis se hizo para ver si el modelo predictivo era mejor que este modelo sencillo.

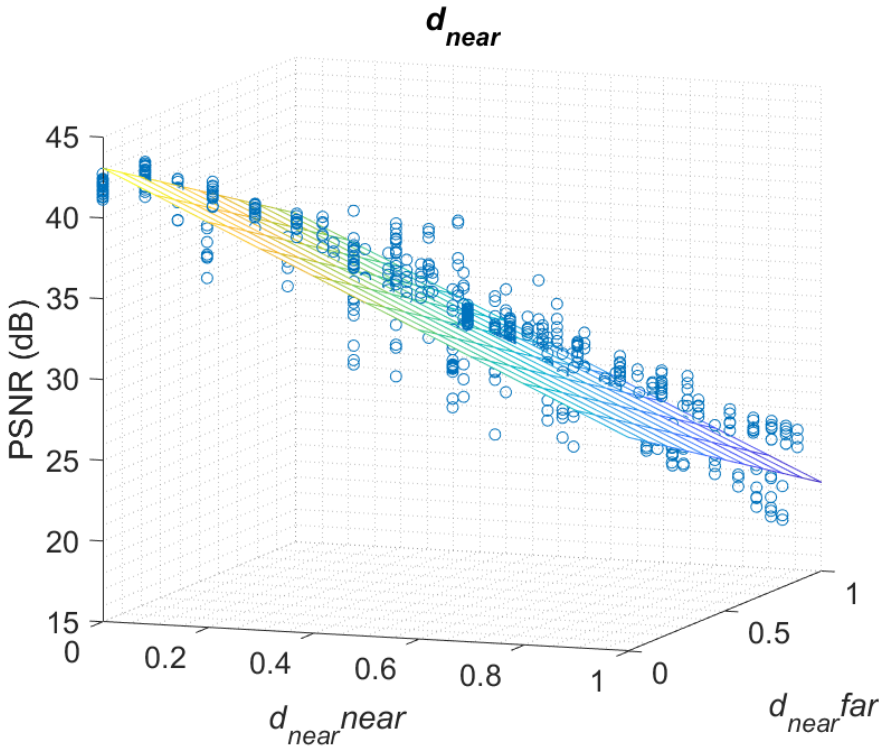


Figura 13: *Representación del valor de PSNR respecto de la disparidad en Znear* [1].

En la Figura 13, vemos que la aproximación mediante el uso de la disparidad máxima de la escena da resultados similares a los propuestos con la disparidad perceptual. El valor del

error de esta aproximación es de 1.42, ligeramente más bajo al de las aproximaciones anteriores mediante la función polinómica y sigmoide. Por tanto, es necesario realizar una prueba con este parámetro también para los nuevos datos, para ver si sigue siendo una alternativa fiable o no.

3.3 Ampliación de los datos de análisis

En esta sección, veremos cómo se han aumentado el número y la variedad de los datos del análisis, tal y como hemos adelantado en la introducción, en el Capítulo 3.1.

3.3.1 Aumento de los escenarios y vistas virtuales

La síntesis de vistas en la primera aproximación del modelo, sobre el cual trabajaremos, se realizó para varios escenarios distintos en los que se iban alternando las vistas de referencia y las virtuales, eliminando en cada caso un número mayor de cámaras de referencia.

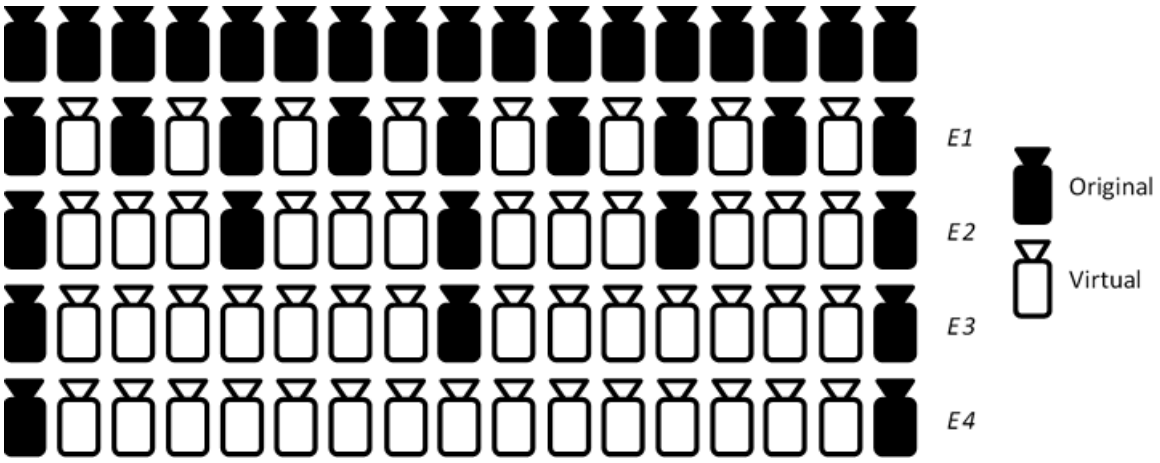


Figura 14: *Escenarios utilizados para el análisis de las secuencias.*

Como podemos observar en la Figura 14, el *Escenario 1* resultó de generar una vista virtual, el *Escenario 2* de generar tres vistas virtuales, el *Escenario 3* de generar siete vistas virtuales, y el *Escenario 4* de generar quince vistas virtuales, todas ellas a partir de dos vistas de referencia originales, situadas en los extremos. Por tanto, la disparidad a las cámaras originales es creciente para cada escenario, obteniendo así cada vez peores medidas de PSNR. Por tanto, para crear estos escenarios, se fueron quitando cada vez más cámaras de referencia, para sintetizar así aquellas vistas virtuales que corresponden al hueco que van dejando las reales a partir de las vistas reales de los extremos.

Para aumentar los datos del análisis, modificaremos el número de cámaras a tener en cuenta. Cambiaremos los escenarios del análisis, de los mostrados en la Figura 14 a una variedad de escenarios mucho mayor. Para un rango y número de cámaras arbitrario, se definirá un escenario de análisis determinado. Para este array de cámaras definido, se generará una vista virtual en cada una de las posiciones de las cámaras, utilizando todas las combinaciones posibles de cámaras de referencia para generarlas.

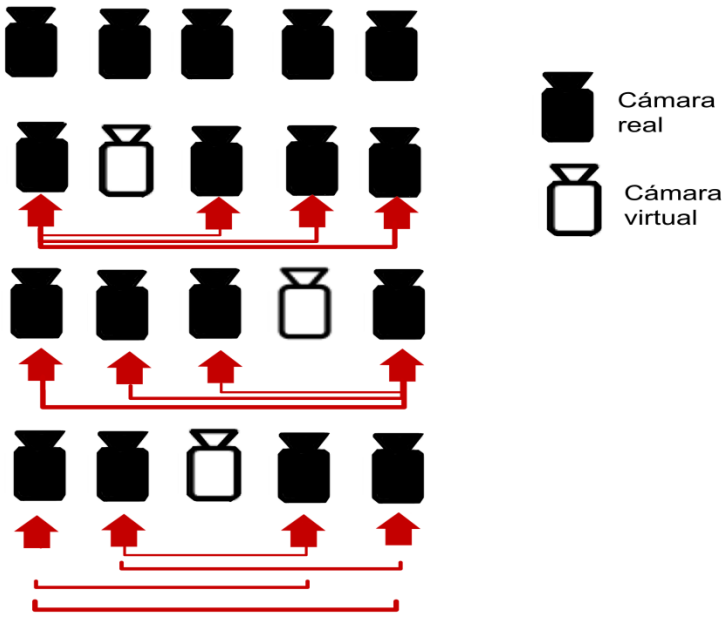


Figura 15: *Combinaciones posibles de síntesis de vistas virtuales para un array de 5 cámaras*

En la Figura 15, podemos ver las distintas combinaciones de cámaras sintetizadas a partir de las cámaras de referencia para un array de cinco cámaras.

3.3.2 Modificación en la extracción de valores de PSNR

En la aproximación al modelo que se da en los trabajos anteriores, se agregó un valor de PSNR para todos los cuadros de la secuencia, pero no se extrae cada uno de estos valores. En su lugar, se hizo una media de todos los valores de PSNR de la secuencia, obteniendo un único valor que fue el utilizado en las pruebas. De esta forma, se están promediando efectos de distorsión en cuadros, que pueden tener valores de PSNR distintos.

Una de las mejoras en el modelo de calidad de vistas virtuales, será el aumento en el número de valores de PSNR que se utilizarán en los datos de entrenamiento para cada secuencia. En lugar de utilizar un valor medio de PSNR por secuencia, habilitaremos la posibilidad de trabajar con hasta un valor de PSNR por cuadro, incluyendo la posibilidad de generar y trabajar con un número de valores deseado, a partir del muestreo de todos los cuadros del video. Gracias a ello tenemos unos datos más precisos, de manera que se pueda hacer un ajuste mejor de los parámetros en la optimización, que explicaremos en el Capítulo 3.4.

Por tanto, para las secuencias analizadas, tendremos la posibilidad de ver la variación de $d_{Perc}near$ y $d_{Perc}Far$, así como de PSNR, en función de la variación temporal de la secuencia. El número de cuadros a tener en cuenta será otro parámetro a elegir por el usuario, junto al rango y número de cámaras explicado en la subsección anterior.

3.3.3 Modificación en el cálculo de $d_{Perc}near$ y $d_{Perc}Far$

En el trabajo anterior de [1], basado en el MVPDM, y como se nombra en el Capítulo 2.2.3, se decide basar el modelo en la relación entre los valores de PSNR y los de disparidad perceptual de cada cámara de análisis. En este caso, partimos de que el modelo toma como información de disparidad más relevante, es decir, de profundidad a los objetos, la disparidad perceptual entre la cámara central del array y su cámara consecutiva. A partir de este valor, se realiza el cálculo de disparidad perceptual entre cada una de las cámaras de referencia y la cámara de la vista generada, multiplicando esta disparidad calculada previamente por el número de cámaras de distancia al que se encuentre cada una de las cámaras reales. Sin embargo, este método es una aproximación, ya que la información de profundidad sobre la que se parte pertenece a la cámara central de la secuencia. Por tanto, es necesario modificar esta forma de calcular los valores de $\boldsymbol{d_{Perc}Near}$ y $\boldsymbol{d_{Perc}Far}$ para ver si este método de cálculo sigue siendo válido para los nuevos datos más precisos. Para mejorar la aproximación de este parámetro, partiremos de la disparidad perceptual entre cada cámara de análisis y su consecutiva, en lugar de utilizar siempre la de la central. Después, la multiplicaremos por el número de cámaras de distancia al que se encuentren. En el Capítulo 4 veremos si la modificación de este cálculo nos lleva a tener una aproximación más precisa que la de los trabajos anteriores.

3.4 Optimización aplicada a las nuevas condiciones

Como hemos visto en el Capítulo 3.2, para el cálculo de la disparidad perceptual, se utiliza una función de peso que dependerá de unos valores de α y β determinados. Después de realizar las pruebas con los nuevos datos de entrenamiento, tendremos que realizar una optimización del modelo para los nuevos datos, para ver cuáles son los nuevos valores de α y β óptimos. El objetivo, será encontrar el plano que mejor aproxime la correlación entre los valores de PSNR, $\boldsymbol{d_{Perc}near}$ y $\boldsymbol{d_{Perc}far}$**.** Para encontrar la mejor aproximación, se aplicará una regresión lineal, donde las variables independientes sean $\boldsymbol{d_{Perc}near}$ y $\boldsymbol{d_{Perc}far}$ y la variable dependiente, los valores de PSNR. Así podremos saber cuál sería el plano óptimo para nuestros datos. De esta forma, se hace esta regresión para cada valor de α y β, minimizando así el error mostrado en (5). De entre todas las combinaciones de α y β, nos quedaremos con el valor del error mínimo de (5). La expresión del error, para un valor N, que indica el número de vistas sintetizadas, es la siguiente.

$$\boldsymbol{Error = \frac{1}{N}\sum_{i=1}^{N} |PSNR_i - z_i|} \qquad \textbf{(5)}$$

Como podemos ver, el error calculado para cada vista i, resulta del módulo de la diferencia entre el valor de PSNR de cada vista sintetizada y el valor de PSNR óptimo predicho a partir de la regresión lineal multivariable, $\mathbf{z_i}$, calculado como $\boldsymbol{z=\beta_0+\beta_1 d_{Perc}near + \beta_2 d_{Perc}far}$.

4 Integración, pruebas y resultados

4.1 Introducción

En esta sección integramos los cambios en los datos explicados en la sección anterior, y veremos qué efecto tienen en el modelo previo de [1]. En primer lugar, hablaremos sobre el contenido utilizado en nuestro análisis. Después, explicaremos y representaremos los resultados de la variación, tanto de los valores de PSNR como de disparidad perceptual a lo largo del espacio y del tiempo para los nuevos datos. Continuaremos explicando el resultado definitivo del modelo de predicción de calidad después de realizar la optimización del mismo, explicada en el Capítulo 3.4. Finalmente, haremos una breve descripción sobre las herramientas que hemos creado para poder extraer y trabajar con una cantidad de datos tan grande de manera eficiente.

4.2 Contenido utilizado

Para el análisis de este modelo utilizaremos 4 secuencias de vídeo SMV. Las secuencias que hemos elegido son utilizadas típicamente por los grupos de trabajo de MPEG para video SMV. En primer lugar, utilizaremos las secuencias *Butterfly* y *Flowers* [14]. Estas dos secuencias de video están capturadas a partir de un array no lineal de cámaras, es decir, las cámaras están situadas en forma de arco. Además, las vistas reales a partir de las cuales sintetizaremos las virtuales, no están capturadas a partir de cámaras reales, sino que están generadas por ordenador, por lo que su información de profundidad será la ideal, y la aparición de artefactos en la síntesis será menor. Por otro lado, se utilizarán las secuencias *Champagne* y *Pantomime* [14], que están capturadas por un array de cámaras lineal (situadas en línea recta) formado por cámaras reales, por lo que su información de profundidad será una estimación y no será tan precisa como la de las dos anteriores, obteniendo a priori más error en la síntesis. Ésta información de profundidad se obtiene a partir de Depth Estimation Algorithms (DERS) [7].

Cámara central secuencia Butterfly

Cámara central secuencia Flowers

Cámara central secuencia Champagne *Cámara central secuencia Pantomime*

Figura 16**:** *Vistas de la cámara central para cada una de las secuencias.*

Contenido	**Disposición de las cámaras**	**Resolución**	**Frame Rate**	**Número de cámaras**	**Z_{near}**	**Z_{far}**
Butterfly	Arco	1280 x 768	24 fps	91	1270 BU	700 BU
Flowers	Arco	1280 x 768	24 fps	80	0.200 BU	595 BU
Champagne	Lineal paralela	1280 x 960	30 fps	80	3222 mm	8215 mm
Pantomime	Lineal paralela	1280 x 960	30 fps	80	4498 mm	8222 mm

Tabla 2*: Características de las secuencias de vídeo SMV utilizadas.*

En la Tabla 2 podemos ver las características que tienen las secuencias de video que utilizaremos en nuestro trabajo.

4.3 Ampliación de los datos: Análisis espacial

En esta sección analizaremos los resultados del modelo tras aplicar el aumento en los datos de entrenamiento, mostrado en el Capítulo 3.3.2, es decir, veremos la variación de la disparidad perceptual y PSNR a medida que se avanza en el tiempo, para ver la variabilidad temporal de los datos en las secuencias utilizadas.

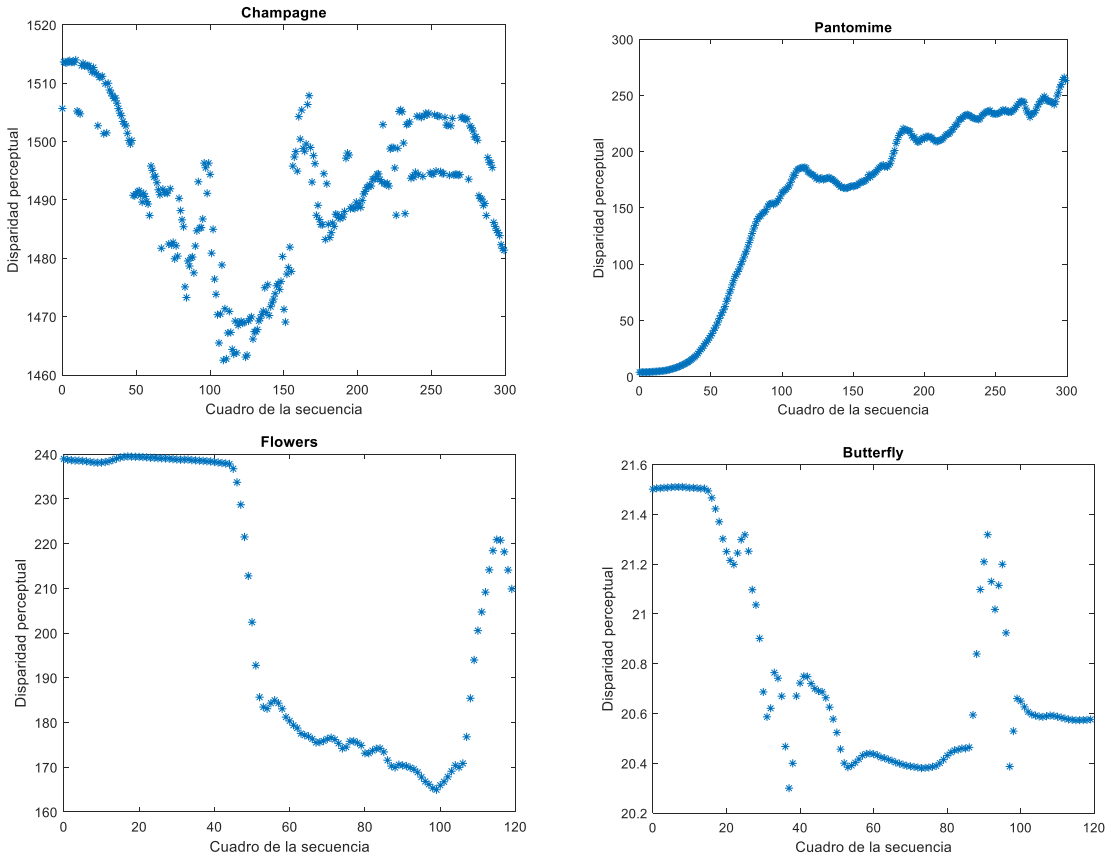


Figura 17**:** *Representación gráfica de los valores de disparidad en función de cada cuadro para las cuatro secuencias utilizadas*

En la Figura 17, podemos ver la variación temporal de la disparidad perceptual entre la cámara central y su consecutiva a lo largo de cada una de las secuencias mostradas en la Figura 16. Para el análisis, se ha elegido la cámara de la posición número 44. Hemos elegido esta cámara ya que es una cámara central para el rango de cámaras de las cuatro secuencias, dado que en las cámaras de los extremos no habrá apenas variación temporal, pues solo se puede observar un extremo de la escena. Como parámetros, hemos escogido valores de $\alpha = 2.5$ y $\beta = 1.25$, siendo éstos los valores óptimos de los resultados del trabajo previo mostrado en [1].

Observamos que para la secuencia *Champagne*, los valores de disparidad varían en un rango que oscila entre 1460 y 1520 sin seguir un patrón determinado. Lo mismo ocurre para la secuencia *Flowers* pero con un ratio de variación que se mueve entre 160 y 240. La secuencia *Butterfly* tiene una variación en la disparidad mucho menor, que varía entre 20.2 y 21.6. En la secuencia *Pantomime* los valores de disparidad perceptual aumentan a medida que avanzamos en el tiempo, con unos valores que van desde 0 hasta 300.

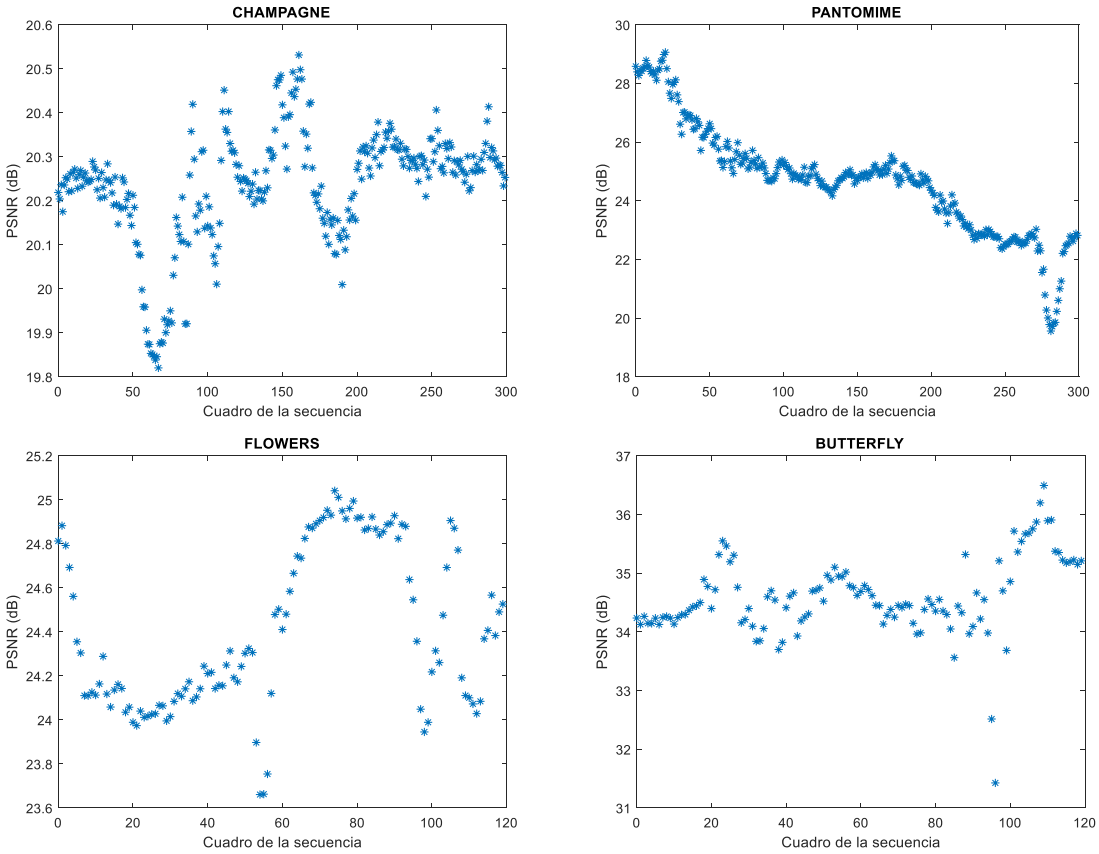


Figura 18*: Representación gráfica de los valores de PSNR en función de cada cuadro para las cuatro secuencias utilizadas.*

En la Figura 18, podemos ver la variación de los valores de PSNR a lo largo del tiempo para cada una de las secuencias. La cámara del análisis será la 44, sintetizada a partir de las vistas de las cámaras 41 y 47, distanciadas tres cámaras de distancia. Para la secuencia *Champagne*, el PSNR no tiene valores muy altos y apenas varía, situándose en torno a los 20 dB. Lo mismo ocurre para la secuencia *Flowers*, pero con valores algo superiores, sobre los 24 dB. En la secuencia *Butterfly* los valores de PSNR oscilan en un rango de valores algo mayores, entre 31 dB y 37 dB, pero sin seguir una variación definida. En la secuencia *Pantomime*, el PSNR roza los 30 dB en los primeros cuadros de la secuencia y disminuye a lo largo de ésta, hasta llegar a los 20 dB.

4.4 Ampliación de los datos: Análisis temporal

Como hemos visto en el Capítulo 3.3.3, hemos modificado la manera en la que se calculaban los valores de disparidad perceptual a la cámara cercana y lejana. Por tanto, pasaremos de tener un valor de disparidad perceptual para cada secuencia, a partir del cual calcular $d_{Perc}near$ y $d_{Perc}far$, a tener tantos valores como cámaras del array se analicen. El objetivo de este capítulo es analizar la variación de los datos usados en la parametrización en función del índice de cámara para las diferentes secuencias, antes de hacer el análisis de los nuevos datos de entrenamiento.

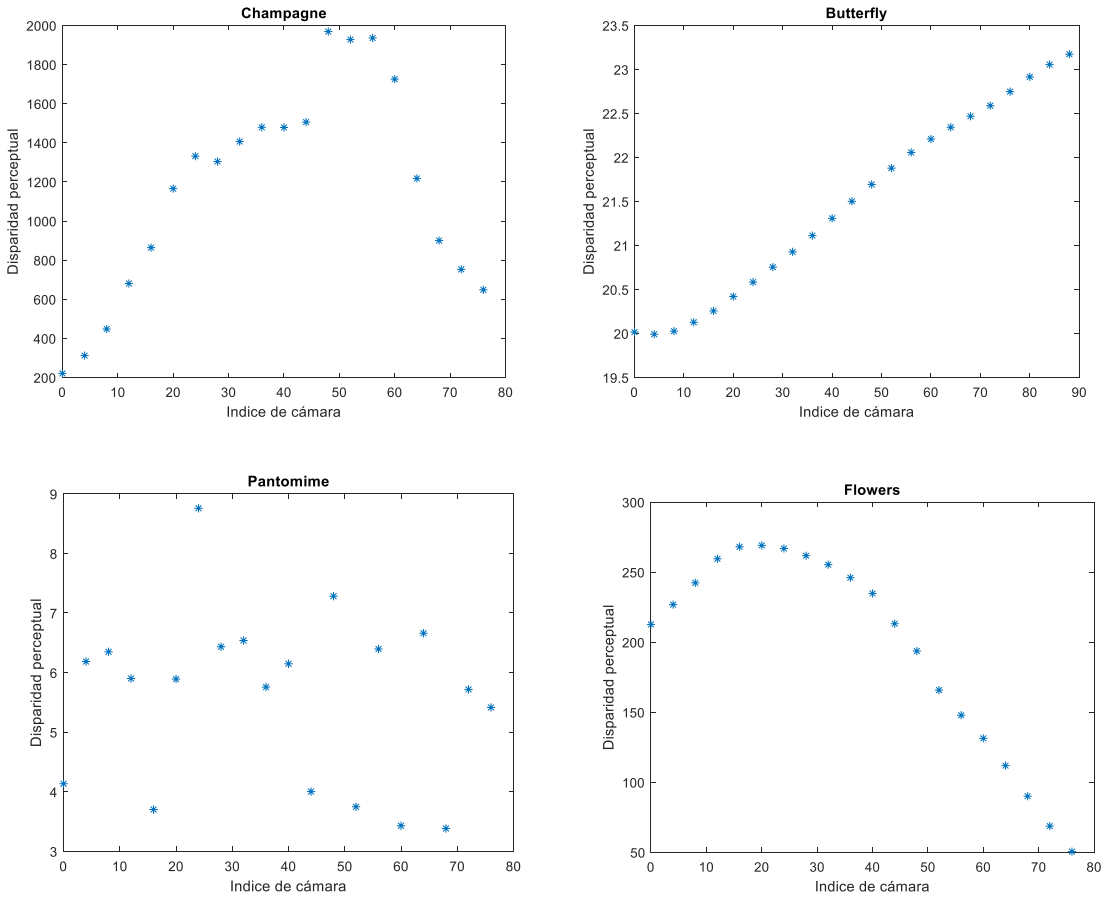


Figura 19*: Representación gráfica de los valores de disparidad perceptual en función del índice de cámara para las cuatro secuencias utilizadas.*

En la Figura 19, podemos ver la variación del valor de disparidad perceptual entre cada una de las cámaras analizadas y su cámara consecutiva, en función del índice de cámara. Para ello, hemos tomado en cuenta el primer cuadro de cada una de las secuencias. Podemos observar, que éstas varían de forma muy distinta, ya que, tanto la disposición de los objetos como los diferentes valores de profundidad de la escena varían considerablemente entre unas y otras secuencias. Por ejemplo, en la secuencia *Champagne*, podemos observar una disparidad muy baja en las vistas correspondientes a los primeros índices, en torno a 200, aumentando después hasta valores que llegan a 2000. Esto se debe a que la cámara situada en el extremo para esta secuencia, no ve los objetos de la cámara más cercanos, que son los que aumentan la disparidad de la escena, como se puede apreciar en la Figura 16.

Además, podemos observar que en las secuencias *Pantomime* y *Butterfly* los valores de disparidad no varían apenas según el índice de cámara, variando entre 4 y 9 para *Pantomime* y entre 20 y 23.5 para *Butterfly*. No ocurre lo mismo para la secuencia *Flowers*, en la que los valores de disparidad van disminuyendo según nos movemos en el array de cámaras, por lo que los objetos más cercanos de la escena se encontrarán en torno al extremo izquierdo. Por tanto, el utilizar solo el valor de disparidad perceptual de la cámara central para calcular $\boldsymbol{d_{perc}near}$ y $\boldsymbol{d_{perc}far}$ en estas secuencias no era un buen método, ya que se estaba utilizando un valor único que en realidad varía para cada una de ellas. Por todo ello creemos que debe tenerse en cuenta, como hemos hecho en la mejora explicada en el Capítulo 3.3.3.

4.5 Resultados del análisis para los nuevos datos

A continuación veremos los resultados finales del modelo propuesto después de la optimización de los parámetros de las funciones de peso y el aumento y mejora de los datos de prueba, explicado en el Capítulo 3. En primer lugar, haremos un nuevo análisis del modelo propuesto en [1] para todo el rango de cámaras de cada secuencia, analizando 20 cámaras submuestradas en el espacio para el caso de las secuencias *Champagne*, *Flowers* y *Pantomime*. Para la secuencia *Butterfly* analizaremos 23 cámaras en lugar de 20, por tener un array de 90 cámaras en lugar de 80, como se muestra en la Tabla 2. En cuanto a la variación temporal, analizaremos únicamente el primer cuadro de cada secuencia. Después, haremos un análisis con condiciones iniciales similares pero utilizando la máxima disparidad de la escena a la cámara cercana y lejana en lugar de $\boldsymbol{d_{perc}near}$ y $\boldsymbol{d_{perc}far}$. Para concluir, haremos un último análisis, esta vez para un rango reducido de cámaras, pero utilizando todos los valores de PSNR en función de $\boldsymbol{d_{perc}near}$ y $\boldsymbol{d_{perc}far}$ para todos los cuadros de cada secuencia. Para este caso, utilizaremos 6 cámaras de análisis submuestreadas entre las posiciones 31 y 49.

4.5.1 Resultados para el análisis espacial

En este capítulo, realizaremos el análisis teniendo en cuenta todo el rango de cámaras. En cuanto a la variación temporal, hemos tenido en cuenta sólo el primer cuadro de cada una de las secuencias, como se muestra en el Capítulo 4.4, ya que donde mayor es la variación de los valores de disparidad es en el espacio. Un ejemplo del conjunto de datos que utilizaremos pero para una cantidad de cámaras y cuadros por secuencia menor, es el que se muestra en la Figura 20.

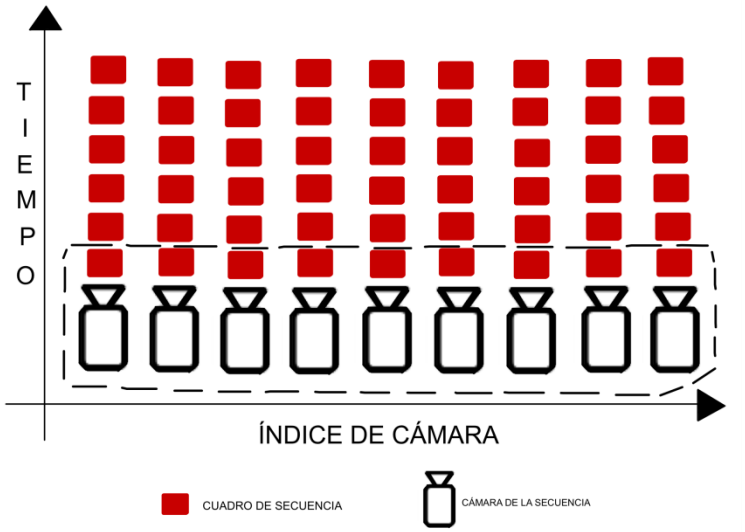


Figura 20: *Ejemplo de análisis espacial para un array de 9 cámaras y 6 cuadros por secuencia*

Para cada una de las cámaras analizadas, se sintetizarán todas las combinaciones posibles de vistas a partir de otras de referencia, tal y como se puede ver en el ejemplo la Figura 15. Utilizaremos 20 cámaras en el caso de las secuencias *Champagne*, *Flowers y Pantomime*, y 23 en el caso de la secuencia *Butterfly*. Las cámaras que analizaremos serán, por tanto, la 0, 3, 7, 11...79 para los arrays de 20 cámaras y la 0, 3, 7, 11...89 para la secuencia *Butterfly*.

Después, calcularemos el PSNR de cada una de las vistas generadas, y realizaremos el cálculo de $\boldsymbol{d_{perc}near}$ y $\boldsymbol{d_{perc}far}$ mediante el método explicado en el Capítulo 3.3.3. Finalmente, para el conjunto de valores de $\boldsymbol{d_{perc}near}$ y $\boldsymbol{d_{perc}far}$ resultante de todas las combinaciones, se buscará el valor óptimo de los valores de α y β de las funciones polinómica (2) y sigmoide (3), mediante la optimización explicada en el Capítulo 3.4. Hemos hecho la optimización para un rango de valores entre -3 y 3 para la función polinómica y entre -1 y 2 para la función sigmoide, con una variación muy pequeña, de 0.1. Para los valores de α se ha estudiado una variación entre 0 y 3 para las dos funciones, con el mismo salto, de 0.1.

En nuestro experimento, al utilizar 20 y 23 cámaras, se han obtenido 969 y 1540 combinaciones de valores de $\boldsymbol{d_{perc}near}$ y $\boldsymbol{d_{perc}far}$ distintas respectivamente para cada valor de α y β. Además, se han utilizado 969 y 1540 valores de PSNR, resultantes de todas las combinaciones de cámaras sintetizadas posibles. Los valores de PSNR en función de los de $\boldsymbol{d_{perc}near}$ y $\boldsymbol{d_{perc}far}$ se mostrarán a continuación, en la Figura 21 para el caso de la función polinómica, y en la Figura 22 para la función sigmoide.

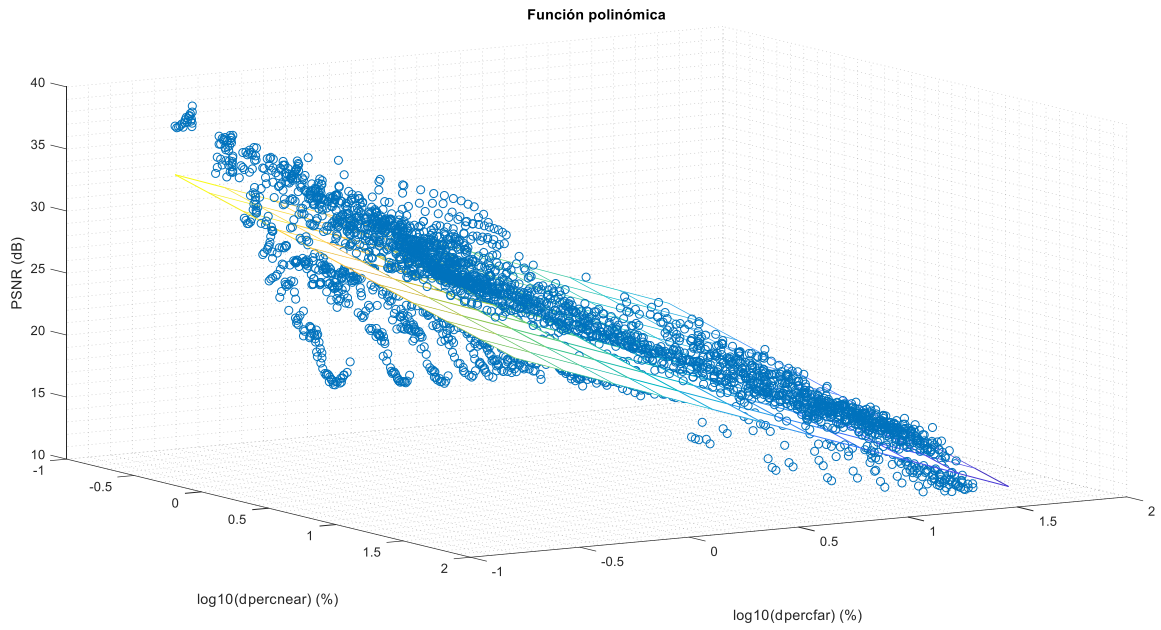


Figura 21*: Aproximación óptima del PSNR a partir de los valores de $d_{perc}near$ y $d_{perc}far$ para la función polinómica.*

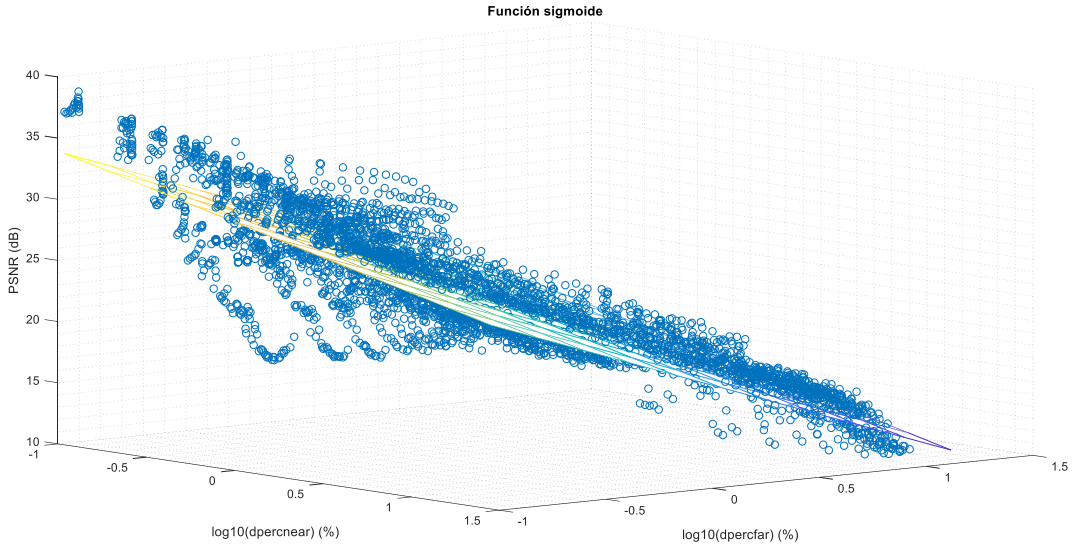


Figura 22*: Aproximación óptima del PSNR a partir de los valores de $d_{perc}near$ y $d_{perc}far$ para la función sigmoide.*

Tabla 3: Valores óptimos de α y β para la función polinómica y sigmoide.

	Función Polinómica	**Función Sigmoide**
α óptimo	**0.9**	**0.6**
β óptimo	**-0.2**	**-0.2**
Error	**1.88**	**1.804**

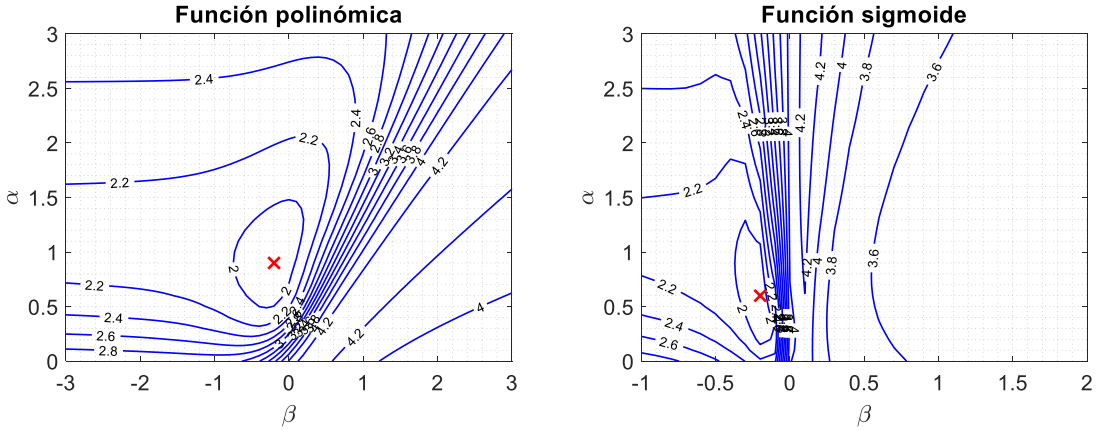


Figura 23: *Resultados del error mínimo medio para un rango de valores de α y β (valor óptimo señalado en rojo)*

Tanto para la función polinómica (2) como sigmoide (3), los resultados son similares. Como se puede apreciar en las Figura 21 y Figura 22, los valores de PSNR de las secuencias probadas no son muy diferentes a los del trabajo previo analizado en [1]. Podemos confirmar, por tanto, que podemos tener una predicción de los valores de PSNR a partir de los de disparidad perceptual. Sin embargo, no obtenemos los resultados deseados en zonas de disparidades bajas de la escena, donde la variabilidad en los valores de PSNR es mucho mayor. En esta zona, para valores similares de disparidad, tenemos mucha variación en los valores de PSNR, lo que significa que no es un buen predictor. Para disparidades mayores de la escena, los valores de PSNR están menos dispersos que para valores de disparidad menor.

Por tanto, para cámaras cercanas la calidad no solo depende de un único valor de disparidad como puede ser $\boldsymbol{d_{perc}near}$, sino que depende de más parámetros que sería interesante incluir en este modelo de cara a líneas futuras.

En cuanto a los valores óptimos de α y β, como se puede ver en la Tabla 3, son bastante parecidos entre una función y otra, llegando a ser iguales para β, pero distan bastante de los valores óptimos que se obtuvieron en los trabajos previos, como se muestra en la Tabla 1.

Las regiones donde se encuentra el valor del error mínimo medio se muestran en la Figura 23, en la que podemos ver que distan bastante de las regiones donde se encontraba el error mínimo medio del trabajo anterior, mostradas en la Figura 12, siendo los valores tanto de α como de β mayores que en nuestro análisis.

El valor de β, como podemos ver, es muy pequeño, de -0.2. Esto se debe a que a pesar de que la disparidad de los objetos por detrás del plano de convergencia sea alta, dado que las texturas de los fondos son muy homogéneas, no generan distorsión en la síntesis, y por tanto disparidades altas.

4.5.2 Resultado para la disparidad en Znear

Como hemos visto en el Capítulo 3.2, el uso de la disparidad máxima de la escena lleva a una buena aproximación de los valores de PSNR en función de ella, por lo que analizaremos estos resultados para los nuevos datos de entrenamiento. En este caso, no solo se utilizará la disparidad máxima de la cámara central, sino que se calcularán todas las combinaciones posibles de valores de disparidad máxima para todas las cámaras analizadas. Esto se hace de forma análoga al cálculo de $\mathbf{d}_{perc}\boldsymbol{near}$ y $\boldsymbol{d}_{perc}\boldsymbol{far}$ que desarrollamos en el Capítulo 3.3.3. Para el análisis de este parámetro se utilizarán las mismas condiciones iniciales que en el Capítulo 4.5.1.

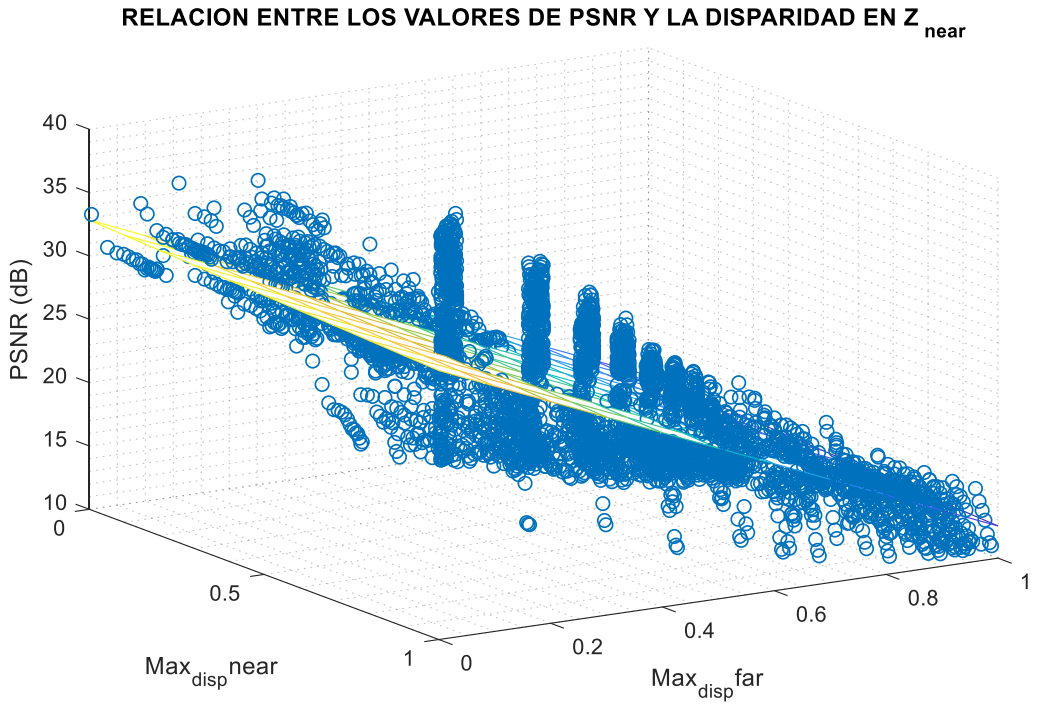


Figura 24*: Aproximación óptima de PSNR a partir de los valores de disparidad máxima de la escena.*

Tabla 4: Comparativa valores de error para d_{perc} y Z_{near}.

	d_{perc} Polinómica	d_{perc} Sigmoide	Z_{near}
Error	**1.88**	**1.804**	**2.91**

En la Figura 24 se puede ver que la aproximación de los valores de PSNR mediante la disparidad máxima de la escena, en este caso, ofrece peores resultados que los que ofrecía para las condiciones iniciales del modelo, y peores de lo que lo hace la aproximación mediante la disparidad perceptual. Observamos que para un mismo valor de disparidad hay un rango de valores muy grande de PSNR. Esto se debe a que solo se tienen en cuenta los valores de disparidad entre cada cámara y su contigua, en lugar del histograma de disparidades que tiene en cuenta la $\boldsymbol{d_{perc}}$, tal y como se define en la ecuación de (1). El error mínimo medio (5) es mayor que el que había en la primera aproximación, y mayor que el de las funciones polinómica y sigmoide de nuestro análisis espacial, como podemos ver en la Tabla 4. A pesar de ser un método ventajoso por su simplicidad, lleva a una peor aproximación que la conseguida con $\boldsymbol{d_{perc.}}$

4.5.3 Análisis para la variación temporal

El objetivo de este capítulo es realizar el análisis del modelo para un rango limitado de cámaras, pero para todos los cuadros de cada secuencia. El concepto general de este análisis se muestra en la Figura 25, en el que se puede ver el conjunto de datos de entrenamiento similar al nuestro para una cantidad de cuadros y cámaras menor.

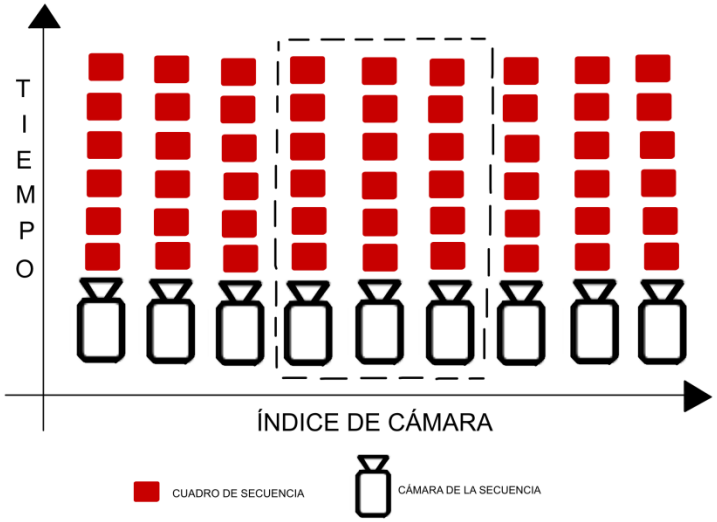


Figura 25: *Ejemplo de análisis temporal para un array de 9 cámaras y 6 cuadros por secuencia.*

Para nuestro experimento, analizaremos las cámaras correspondientes a las posiciones 34, 37, 40, 44, 47 y 49 para todas las combinaciones posibles de vistas sintetizadas. Tal y como en el análisis anterior, pero para los 120 cuadros de las secuencias *Butterfly* y *Flowers*, y para los 300 cuadros de las secuencias *Champagne* y *Pantomime*. Los parámetros de las funciones de peso utilizados serán los óptimos del análisis anterior, de α= 0.9 y β= -0.2 para la función polinómica, y de α = 0.6 y β = -0.2 para la función sigmoide, como se muestra en la Tabla 3. El objetivo será ver cómo se ajustan los datos de entrenamiento, utilizando toda la variación temporal de la secuencia para las condiciones óptimas del primer análisis mostrado en 4.5.1.

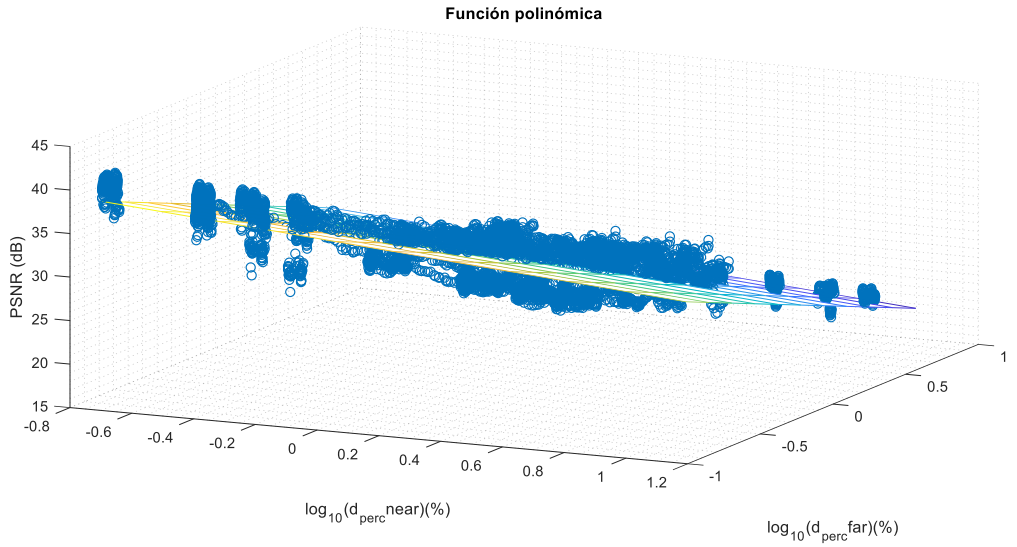


Figura 26: *Representación de los valores de PSNR en función de $d_{perc}near$ y $d_{perc}far$*

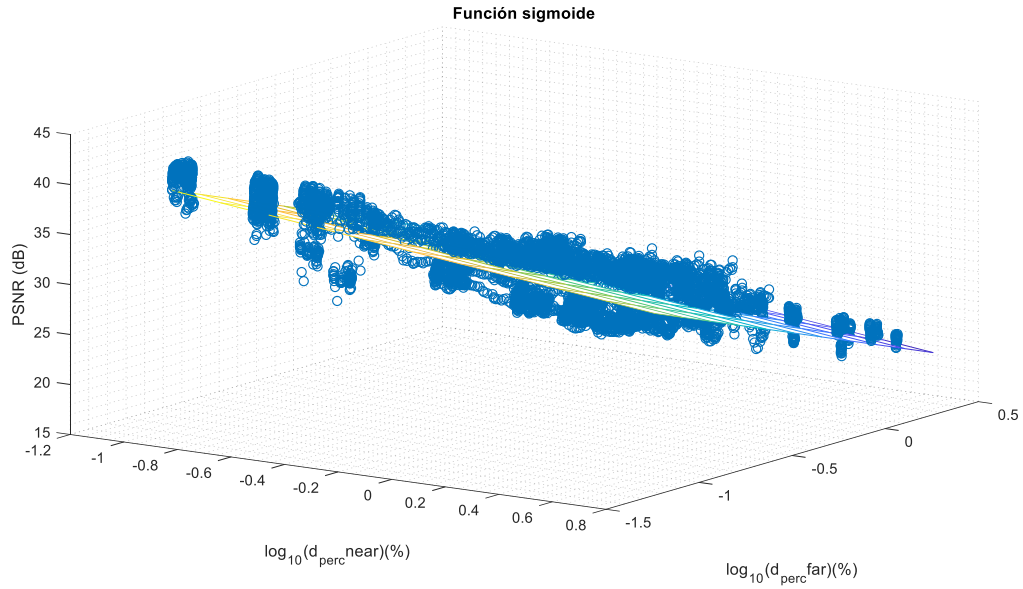


Figura 27: *Representación de los valores de PSNR en función de $d_{perc}near$ y $d_{perc}far$*

Tabla 5: Valores de error para las dos funciones de peso.

	d_{perc} Polinómica	**d_{perc} Sigmoide**
Error	**1.564**	**1.5251**

Los resultados de esta aproximación de los valores de PSNR en función de $\boldsymbol{d_{perc}}$, mostrados en la Figura 26, son mejores a los del análisis anterior. Como podemos ver, el plano que aproxima los valores de PSNR se asemeja más al óptimo que en el análisis espacial. Además, el error es menor, lo que nos da una idea de que, utilizando todas las cámaras y todos los cuadros de la secuencia, la aproximación podría mejorar.

En el caso de la función sigmoide, los resultados de la Figura 27 son los esperados, similares a los de la función polinómica, pero con un valor del error algo menor, por lo que sigue siendo una aproximación algo mejor que la de la función polinómica.

Por tanto, podemos decir que la parametrización es lo suficientemente general como para predecir valores de calidad a partir de la disparidad perceptual para contenidos de la escena que han variado a lo largo del tiempo.

El siguiente paso, sería realizar la optimización mostrada en el Capítulo 3.4 para estos nuevos datos, para ver si los parámetros óptimos son muy diferentes a los utilizados.

4.6 Herramienta automática de generación de datos

Si tenemos en cuenta los datos utilizados en la mejora de la aproximación, vemos que tanto la síntesis de vistas virtuales sintetizadas y los cálculos del valor de PSNR por cuadro de cada una de ellas, requieren mucho tiempo, tanto de configuración en los programas utilizados para ello, como en la ejecución. Por tanto, para facilitar este trabajo y poder realizar pruebas con cualquier cantidad y tipo de datos, hemos desarrollado una herramienta de generación automática de los datos, a partir de la parametrización de todas las variables que entran en el proceso. El algoritmo se ha desarrollado en Python por ser un lenguaje de programación extendido y sencillo de utilizar.

Este programa debe ser paralelo al análisis explicado en las secciones anteriores, ya que debe generar los datos exactos que utilizaremos después para la aproximación de los valores de PSNR a la disparidad perceptual, tanto para el análisis temporal (variación por cuadro) como espacial (distintas combinaciones de cámaras). Para generar los datos se seguirán los siguientes pasos:

1. Se define la configuración de las cámaras eligiendo la cámara del extremo izquierdo y derecho del array, así como el número de cámaras que se quiere analizar. A partir de estos datos, se calcula un salto entre cámaras para saber qué cámaras submuestreadas se utilizarán.

2. Síntesis de cada una de las vistas resultantes de la combinación de cámaras posible, tal y como se muestra en el ejemplo de la Figura 15. Para ello, se utilizará el programa VSRS.

3. Cálculo del valor de PSNR de cada vista virtual sintetizada respecto de la cámara original en esa posición.

Después, para el análisis de los datos de entrenamiento, se han creado herramientas en Matlab, así como código utilizado en el trabajo de [1] y en el MVPDM, que hemos adaptado para el nuevo análisis.

5 Conclusiones y trabajo futuro

El objetivo principal de este trabajo de fin de grado era ampliar los datos del análisis que se llevó a cabo en [1], ya que sus datos de entrenamiento no utilizaban toda la información espacial y temporal de la escena, sino que utilizaban agregaciones de los datos. El trabajo previo propone un modelo que permite predecir valores de calidad objetiva de secuencias de video SMV a partir de determinadas características de la escena. Nuestro objetivo principal era ver cómo generalizaba este modelo para un análisis con un número y variedad de datos mayor, para el que hemos sacado las siguientes conclusiones.

En primer lugar, los resultados obtenidos para las nuevas condiciones del análisis, en el que se estudia la variación espacial en todo el rango de cámaras, son similares a los obtenidos en el trabajo previo. Sin embargo, observamos que para las zonas de disparidad de la escena baja, es decir, para el caso de disparidad entre cámaras cercanas, los valores de PSNR son mucho más dispersos, y no dependerán únicamente de la disparidad perceptual de la escena. Esto se puede apreciar en la Figura 21. Por tanto, sería interesante incluir otros parámetros en el análisis de este modelo, que tengan en cuenta características de la textura de la imagen, y no solo de la profundidad, para ver si mejora la predicción de los valores de PSNR en esas regiones.

Tras la optimización de los valores de α y β, se observa que tenemos unos valores de β mucho menores. Esto se debe, a que, aunque la disparidad sea alta, si las texturas del fondo son muy homogéneas, no generarán error en la síntesis, y la disparidad calculada no será la real. Por tanto, una buena línea de trabajo futuro será probar este análisis para secuencias de video en las que el fondo tenga texturas más complejas.

Para el análisis mediante la disparidad máxima de la escena, no obtuvimos resultados similares a los del trabajo previo. En el modelo previo, este parámetro sirvió para obtener una buena aproximación de los valores de PSNR, además de ser una parametrización mucho más simple. Sin embargo, para nuestros nuevos datos, el error en la predicción fue mayor al obtenido en el análisis mediante la disparidad perceptual, por lo que creemos que una parametrización más elaborada da mejores resultados.

En el análisis para el que se utilizan todos los cuadros de la secuencia, la aproximación de los valores de PSNR es más precisa que para el análisis espacial, obteniendo valores de error de predicción menores, por lo que el modelo es válido para contenidos de la escena que varían a lo largo del tiempo. Es decir, el modelo generaliza bien para datos diferentes a los usados en el primer entrenamiento.

En un futuro, es conveniente utilizar medidas de calidad subjetiva para entrenar el modelo, en lugar de medidas de calidad objetiva, a pesar del gasto de recursos que ello supone.

Por último, en nuestro trabajo se han utilizado las mismas secuencias que se utilizan en [1], por lo que una posible línea de trabajo futuro a corto plazo, sería la de entrenar el modelo para los nuevos datos, pero utilizando un número de secuencias mayor.

Referencias

[1] Miranda Espallargues, Diana (2018). Diseño de un modelo de calidad de vistas virtuales para vídeo multivista. Proyecto Fin de Carrera / Trabajo Fin de Grado, E.T.S.I. Telecomunicación (UPM).

[2] Fernández Rivero, Juan Antonio (2004). Tres dimensiones en la historia de la fotografía: La imagen estereoscópica. Málaga: Miramar. ISBN 9788493209452.

[3] A. Smolic, "3D video and free viewpoint video – From capture to display", *Pattern Recognition*, vol. 44, Issue 9, pp. 1958-1968, ISSN 0031-3203, Sep. 2011.

[4] G. Balota, M. Saldanha, G. Sanchez, B. Zatt, M. Porto and L. Agostini, "Overview and quality analysis in 3D-HEVC emergent video coding standard," *IEEE 5th Latin American Symposium on Circuits and Systems*, Santiago, pp. 1-4. May. 2014.

[5] Pereira, Fernando & da Silva, Eduardo & Lafruit, Gauthier. (2018). Plenoptic imaging: Representation and processing. 75-111. 10.1016/B978-0-12-811889-4.00002-6.

[6] P. Carballeira, J. Gutiérrez, F. Morán, J. Cabrera, F. Jaureguizar, N. García, "Multiview Perceptual Disparity Model for Super Multiview Video", *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 1, pp. 113-124, Feb. 2017.

[7] M. T. O. Stankiewicz, K. Wegner and M. Domanski, "Enhanced Depth Estimation Reference Software (DERS) for Free-viewpoint Television," Contrib. M31518, 106th MPEG meeting, Geneva, CH,Oct. 2013.

[8] S. Winkler, P.Mohandas, "The Evolution of Video Quality Measurement: From PSNR to Hybrid Metrics", *IEEE Transactions on Broadcasting*, vol 54, no. 3, pp. 660-669, Sep. 2008.

[9] M. H. Pinson, S. Wolf, "A New Standardized Method for Objectively Measuring Video Quality", *IEEE Transactions on Broadcasting*, vol. 50, no. 3, pp. 312-322, Sep. 2004.

[10] F. Battisti, E. Bosc, M. Carli, P. Le Callet, S. Perugia, "Objective Image Quality Assessment of 3D Synthsized Views", *Elsevier Signal Processing: Image Communication*, vol. 30, issue C, pp. 78-88, Ene. 2015.

[11] S. L. P. Yasakethu, C. T. Hewage, W. A. C. Fernando, A. M. Kondoz, "Quality analysis for 3Dvideo using 2D video quality models", *IEEE Trans. Consum. Electron*., vol. 54, no. 4, pp.1969-1976, Nov. 2008.

[12] Y. Takaki, "Development of super multi-view displays", *ITE Trans. Media Technol.Appl*., vol. 2, no. 1, pp. 8-14, Jan. 2014.

[13] Sun, W., Xu, L., Au, O. C., Chui, S. H., and Kwok, C. W, “An overview of free view-point depth image-based rendering (DIBR)”, *APSIPA Annual Summit and Conference*, pp. 1023-1030, 2010

[14] ISO/IEC JTC1/SC29/WG11, “Call for Evidence on Free-Viewpoint Television: Super-Multiview and Free Navigation”, Output doc. N15733, 113th MPEG meeting, Geneva, CH, Oct. 2015

[15] A. T. Hinds, D. Doyen, P.Carballeira, G. Lafruit, “Toward the Realization of Six Degrees-offreedom with Compressed Light Fields”, *IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1171-1176, ISSN: 1945-788X, Jul. 2017