

APLICACIÓN DE LÓGICA DIFUSA EN SISTEMAS DE CAPTURA DEL MOVIMIENTO

Carlos Gonzalez-Morcillo, Luis Jiménez
E. Superior de Informática
Universidad de Castilla-La Mancha
Paseo de la Universidad 4, Ciudad Real
{Carlos.Gonzalez, Luis.Jimenez}@uclm.es

Juan Moreno-Garcia
E.U.I. Técnica Industrial
Universidad de Castilla-La Mancha
Avda. Carlos III s/n, Toledo
Juan.Moreno@uclm.es

Resumen

Las capacidades descriptivas de los sistemas de reglas basados en Lógica Difusa ofrecen gran facilidad a la hora de modelar el conocimiento de un experto. Dentro del ámbito de los sistemas de captura del movimiento, ampliamente utilizados en la generación de animaciones con personajes digitales, se suelen emplear modelos basados en curvas de interpolación para la corrección de los diferentes tipos de errores con los que cuentan. Por otro lado, la identificación de este tipo de errores suele realizarse manualmente, con el esfuerzo que esto supone. En este trabajo se presenta un método de identificación de errores en sistemas de captura del movimiento, utilizando un modelo cualitativo, con sencilla validación por parte de un experto. Para finalizar, se muestra un ejemplo desarrollado con el método propuesto junto con los resultados obtenidos.

Palabras Clave: Captura Movimiento, Corrección Errores, Sistemas Cualitativos, Razonamiento Aproximado, Variables Difusas.

1 INTRODUCCIÓN

Habitualmente suele emplearse la lógica difusa en aplicaciones de control [7], [8] y sistemas de tipo Mamdani [3]. Sin embargo, con menor frecuencia se emplea por sus propiedades y facilidades descriptivas [11]. En este trabajo se aprovechan estas características de los sistemas basados en lógica difusa [9] para el modelado de un conjunto de sistemas de corrección de errores en captura del movimiento.

En cualquier producción de animaciones 3D con personajes digitales se pueden distinguir dos fases principalmente; por un lado, la construcción y definición de los elementos que intervienen en la escena (personajes, escenario, etc...) y sus propiedades (iluminación, texturas, cámara, etc...), junto con la animación de los mismos.

En animación, se han desarrollado dos alternativas de trabajo de *alto nivel* frente a las técnicas clásicas de animación basadas en fotogramas clave. Mediante **sistemas de huesos**, el animador puede establecer la posición y rotación de cada elemento de la cadena, afectando a los huesos cuyo nivel de jerarquía sea inferior. Aunque se cuente con cálculo automático de cinemática inversa, este tipo de sistemas requieren un esfuerzo importante por parte del animador, que debe especificar, al menos, la posición de los extremos de las cadenas de huesos en cada keyframe. Por esta razón, se utilizan sistemas de **captura del movimiento**, que captan el movimiento de un actor por medio de unos sensores en el mundo real, aplicándolo posteriormente a los personajes virtuales. Los sistemas de captura generan información acerca de la posición tridimensional de las marcas en cada frame.

1.1 SISTEMAS DE CAPTURA DEL MOVIMIENTO

Existen varios tipos de sistemas de captura del movimiento. Entre los más utilizados están los **sistemas de captura magnéticos**, basados en campos magnéticos y antenas como sensores y los **sistemas de captura ópticos** que utilizan varias cámaras para realizar el seguimiento de unas marcas situadas en el cuerpo del actor.

Los sistemas de captura magnéticos presentan distorsiones en presencia de campos magnéticos y trabajan con un número limitado de marcas. Sin embargo, presentan menos problemas a la hora de identificar las marcas y pueden ser utilizados casi sin procesamiento

posterior, por lo que son utilizables en aplicaciones interactivas. Los sistemas de captura óptica (como el mostrado en la figura 1), no presenta limitación en cuanto al número de marcas y suelen emplearse en la captura de movimientos complejos.

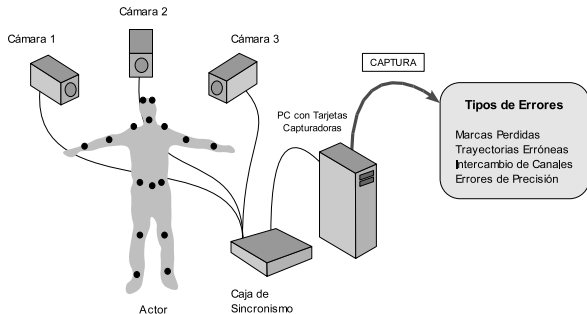


Figura 1: Sistema de Captura Óptico

1.2 PROBLEMÁTICA

La captura del movimiento realizada con sistemas ópticos presenta una serie de problemas (representados en la figura 2). En la figura 2, el eje de abcisas representa el tiempo, y el eje de ordenadas la posición de la marca en el espacio. Un tipo de problema se debe a la **oclusión de marcas** por el propio cuerpo del actor u otros elementos de la captura. Si el actor durante la captura oculta alguna marca en zonas donde al menos dos cámaras¹ puedan verla simultáneamente, el sistema generará errores puntuales en los frames donde la marca no esté visible.

Otro tipo de error muy común es la mala identificación de marcas por parte del sistema de reconstrucción 3D. Partiendo del conjunto de imágenes 2D capturadas, el sistema en algunas ocasiones no asigna las coordenadas 3D reales a una marca, volviendo a tomar de nuevo la trayectoria correcta unos frames más adelante. Esto nos provoca un efecto de **trayectorias erróneas** en ciertos fragmentos de la captura.

Dos marcas que estén próximas en un momento determinado, pueden intercambiar sus valores en un intervalo de tiempo, dándose lugar el tercer tipo de errores de **intercambio de canales**. Estos errores del sistema se deben a una identificación incorrecta de las marcas, y son difíciles de localizar automáticamente de forma general.

Por último, existe un tipo de error por **falta de precisión** que provoca un efecto de *trayectorias temblorosas* en la captura de forma global. Es, en parte, debido a la curvatura de la lente, la mala fijación de

¹Son necesarias dos cámaras simultáneas para poder realizar visión estereoscópica

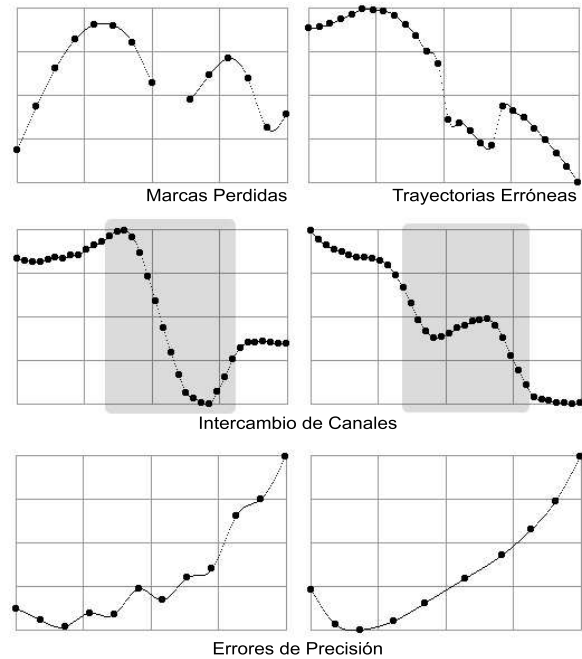


Figura 2: Tipos de errores en Sistemas Ópticos

las marcas reflectantes a las articulaciones del actor y a la pérdida de exactitud en la identificación de puntos 3D.

La detección de estos tipos de errores suele realizarse de forma manual. Un experto reproduce los datos capturados y analiza los diferentes canales de entrada, identificando los problemas que han ocurrido en la fase de captura. Hecho esto, se procede a una fase de limpieza de los datos.

La limpieza de los datos suele realizarse bien de forma semiautomática, empleando interpolación mediante curvas [1], o bien manualmente, situando las marcas erróneas en una posición válida para cada frame.

Esta identificación de errores y su posterior limpieza es una labor muy costosa en tiempo si se realiza de forma manual o con procesos semiautomáticos, llegando a ser el verdadero cuello de botella en los sistemas de captura del movimiento.

2 USOS DE LÓGICA DIFUSA

Nuestra principal motivación es el modelado de un sistema capaz de identificar los errores en flujos de captura de movimiento tal y como lo hace un experto. Para ello, emplearemos sistemas de producción conceptuales basados en lógica difusa, que modelen de forma cualitativa el conocimiento experto. Esto nos permitirá una construcción y validación sencilla de

nuestro sistema de corrección y detección de errores.

Como se ha expuesto anteriormente, hay dos problemas principales al trabajar con sistemas de captura ópticos; por un lado la detección de los errores en cada canal (de algún tipo de los vistos en el apartado 1.2), y por otro la corrección de estos errores. Habitualmente estos procesos suelen realizarse de forma manual. A continuación veremos cómo incorporar detección y corrección de errores de forma automática utilizando lógica difusa.

2.1 DETECCIÓN DE ERRORES

Nuestra intención es construir sistemas difusos especialistas en cada uno de los tipos de errores descritos anteriormente. Así, tendremos un sistema de reglas encargado de localizar errores de marcas perdidas, otro para gestionar trayectorias erróneas, etc. Esto nos permitirá detectar automáticamente los errores en la captura y, generando tal y como lo haría un experto, los parámetros necesarios para proceder a una corrección de errores automática empleando técnicas de interpolación.

Sin embargo, para la detección de intercambios de marcas, necesitamos conocer la configuración jerárquica de las articulaciones que forman nuestro modelo. Esto implica la necesidad de poder calcular la cinemática directa e inversa de nuestro sistema. Tal y como se describe en [5], es posible calcular de manera eficiente la cinemática inversa de estructuras complejas mediante el uso de sistemas de reglas basados en lógica difusa y redes neuronales. Este último aspecto queda fuera del ámbito presente trabajo.

2.2 CORRECCIÓN DE ERRORES

La representación de una marca moviéndose a lo largo del tiempo produce curvas que pasan por un conjunto de puntos de control. Estas curvas pueden gestionarse de forma cómoda mediante polinomios cúbicos, que además no presentan discontinuidades ni ruido en los extremos del intervalo². Así, podemos utilizar Splines Cúbicas Naturales [6] como método de interpolación de curvas para reparar gran parte de los errores detectados en el paso anterior.

Sería posible, además, incorporar filtros de corrección que incorporaran cierto grado de libertad a la hora de obtener un conjunto de trayectorias probables de una marca. Las soluciones cinemáticas obtenidas mediante splines dan resultados buenos, pero no son los únicos posibles. Se podrían tener en cuenta otros factores calculados (velocidades, aceleraciones, etc...) para obtener soluciones particulares para una configuración

²Este problema se conoce como fenómeno de Runge

determinada de esos parámetros. Esta solución necesitaría de la construcción de un sistema de reglas que modelara la salida de un parámetro de tensión (empleando otro tipo de curvas de interpolación, como splines de hermite) con el que obtendríamos movimientos que generaran, por ejemplo, una mayor velocidad de movimientos, una menor aceleración, etc...

3 DESARROLLO DEL SISTEMA

Para la experimentación, validación y prueba de todas las propuestas del presente trabajo, se ha construido una aplicación denominada Pin8 [2], que utiliza un conjunto de sistemas de reglas difusos para la detección de los diferentes tipos de errores existentes en las capturas, donde se han implementado los mecanismos necesarios para la incorporación del conocimiento experto, así como herramientas para la corrección y visualización 3D, como puede verse en la figura 3. Esta aplicación permite la incorporación de nuevos sistemas de reglas difusos mediante la edición de ficheros XML, gestionando en tiempo de ejecución la incorporación de nuevos conjuntos difusos [10] que no estuvieran definidos previamente.

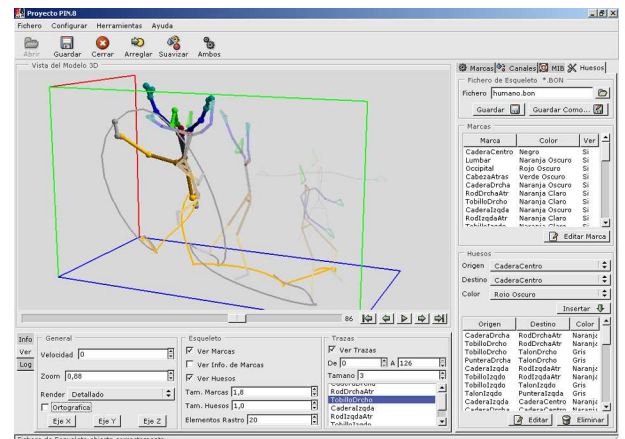


Figura 3: Interfaz de la aplicación

3.1 ANÁLISIS DEL DOMINIO DEL PROBLEMA

En nuestra primera aproximación al problema, nos vamos a centrar en la definición de un conjunto de sistemas que modelen el conocimiento experto en identificación de errores sobre unos canales de entrada. Esto, nos dará un conjunto Ξ de sistemas de reglas. Cada sistema de reglas estará definido sobre su propio conjunto de variables de entrada y sus dominios de definición. Tendremos distintas variables con las que construir nuestros sistemas σ como por ejemplo,

la posición de una marca s en un instante i , que notaremos como $P_{s,i}^\sigma$.

En la identificación de errores, nos hace falta conocer la posición de las marcas en el instante actual t , y en una serie de instantes anteriores t_{i-n} , siendo n el número de pasos de histórico que conservaremos. De esta forma, tenemos $P_{s,t-i}^\sigma = [x_{t-i}, y_{t-i}, z_{t-i}] \forall \sigma \in \Xi, i = 1..n$ en cada coordenada cartesiana.

Detectamos cambios en la velocidad de movimiento de las marcas mediante el conjunto de variables $V_{s,t-i}$ definidas como $V_{s,t-i}^\sigma = \Delta(P_{s,t-i}) = P_{s,t-i}^\sigma - P_{s,t-i-1}^\sigma \forall \sigma \in \Xi, i = 1..n$. De igual forma, nos interesa detectar cambios en la aceleración de las marcas definiendo un conjunto de variables $A_{s,t-i}$ definidas como $A_{s,t-i}^\sigma = \Delta(V_{s,t-i}) = V_{s,t-i}^\sigma - V_{s,t-i-1}^\sigma \forall \sigma \in \Xi, i = 1..n$.

Una vez definidas las variables y los sistemas de reglas, éstos permanecerán sin cambios siempre que utilicemos el mismo tipo de capturas y en el mismo rango de movimientos. Naturalmente, si necesitamos realizar capturas con otro tipo de actor o en otro tipo de problemas, habrá que cambiar el dominio de definición de cada variable difusa (pero no su distribución).

3.2 DEFINICIÓN DEL CONJUNTO DE REGLAS

A continuación vamos a definir los sistemas de reglas que hemos utilizado en la detección de errores en los canales de captura.

3.2.1 Detección de Marcas Perdidas

El primer sistema identifica los errores de marcas perdidas. Estos errores puntuales, en nuestro sistema de captura, vienen representados por un valor máximo que no se alcanza normalmente. Esto se traduce en un cambio muy grande puntual en la velocidad de la marca. Detectaremos cada intervalo que contenga marcas perdidas teniendo en cuenta estos cambios de velocidad. Así, nuestro primer sistema vendrá definido por V_t y V_{t-1} como variables de entrada y E como variable de salida, representando la posibilidad de error en el intervalo.

De este modo, nuestras variables de velocidad definidas para la marca s en el sistema σ , y el tiempo t (con un grado de histórico i), quedan representadas mediante cinco conjuntos difusos: MP (Muy Pequeña), P (Pequeña), N (Nula), M (Media) y G (Grande). Esta notación de los conjuntos difusos se conservará a lo largo del trabajo, aunque varíe el dominio de definición. De esta forma, tenemos, por ejem-

plo, la definición de los conjuntos difusos

$$MP = \int_{-10}^{-9} 1/x + \int_{-9}^{-8} -8 - x/x$$

$$P = \int_{-7}^{-6} x + 7/x + \int_{-6}^{-4} 1/x + \int_{-4}^{-1} \left(\frac{-1-x}{3}\right)/x$$

$$N = \int_{-2}^0 \left(\frac{x+2}{2}\right)/x + \int_{-4}^{-1} \left(\frac{2-x}{2}\right)/x$$

$$M = \int_1^4 \left(\frac{x-1}{3}\right)/x + \int_4^6 1/x + \int_6^8 \left(\frac{8-x}{2}\right)/x$$

$$G = \int_7^9 \left(\frac{x-7}{2}\right)/x + \int_9^{10} 1/x$$

Las variables lingüísticas de entrada se definen sobre el conjunto de etiquetas $\{MP, P, N, G, MG\}$ y la variable de salida sobre $\{P, M, G\}$. Obtenemos el siguiente conjunto de reglas:

- R_1 : IF V_t is MG and V_{t-1} is MP THEN E is G
- R_2 : IF V_t is MG and V_{t-1} is P THEN E is G
- R_3 : IF V_t is MG and V_{t-1} is N THEN E is M
- R_4 : IF V_t is MG and V_{t-1} is G THEN E is P
- R_5 : IF V_t is G and V_{t-1} is MP THEN E is G
- R_6 : IF V_t is G and V_{t-1} is P THEN E is G
- R_7 : IF V_t is G and V_{t-1} is N THEN E is M

Sólo nos interesa conocer el comienzo de cada intervalo de error. La implementación del algoritmo de corrección de errores, basado en Splines Cúbicas Naturales, sólo necesita conocer el punto de comienzo de la interpolación. El punto final será calculado automáticamente. Por esta razón, no se han incluido el conjunto de reglas con valores MG para la variable V_{t-1} .

3.2.2 Detección de Trayectorias Erróneas

Un segundo sistema se encarga de la detección de trayectorias erróneas. Para ello, se utilizan variables de entrada de aceleración. Una aceleración *grande* (ya sea positiva o negativa) en la trayectoria de una marca implica una desviación de la trayectoria original. Esta desviación es signo de trayectoria errónea. Mediante el siguiente sistema podemos localizar, además, los límites del intervalo de frames en el que se produce el error. Las variables de entrada (A_t y A_{t-1}) se definen sobre el conjunto de etiquetas $\{MP, P, N, G, MG\}$. La variable de salida D (Desviación), sobre $\{P, M, G\}$.

Como se ha comentado anteriormente, el dominio de definición de cada variable es dependiente de cada tipo de captura. Un ejemplo de disposición de valores puede verse en la figura 4.

De esta forma, el sistema de reglas empleado en la detección de trayectorias erróneas estará formado por el siguiente conjunto de reglas:

- R_1 : IF A_t is MG and A_{t-1} is N THEN D is G
- R_2 : IF A_t is G and A_{t-1} is N THEN d is M
- R_3 : IF A_t is MP and A_{t-1} is N THEN D is G
- R_4 : IF A_t is P and A_{t-1} is N THEN D is M

Obviamente, R_1 y R_2 servirán para detectar un extremo del intervalo, mientras que R_3 y R_4 detectarán el otro intervalo. Una vez detectado el intervalo de error, se empleará una variación en el método de interpolación de Splines Cúbicas Naturales del sistema anterior para la corrección de los errores.

3.2.3 Detección de Trayectorias Temblorosas

El último sistema utilizado se encarga de la detección de trayectorias temblorosas. Este tipo de detección nos interesa que se realice a nivel global. No es importante detectar en qué intervalo se produce, ya que es un fenómeno general de la marca. Podría desarrollarse un método de asignación del factor de suavizado de la trayectoria dependiente del valor obtenido como salida en la desborrosificación. No obstante, se ha utilizado un valor fijo que da buenos resultados en todas las ocasiones para las capturas realizadas.

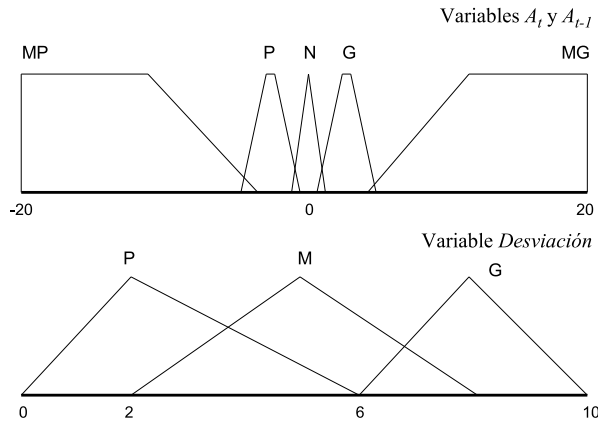


Figura 4: Sistema de detección de Trayectorias Erróneas

En este sistema, las variables de entrada elegidas son $V_t, V_{t-1}, V_{t-2}, V_{t-3}$ definidas sobre las etiquetas $\{P, V\}$ (valores positivo y negativo). La variable de salida S , indicará si el canal necesita suavizado mediante las etiquetas $\{P, M, G\}$. Se necesitará

suavizado si en algún punto de la curva se genera una trayectoria con "dientes de sierra".

- R_1 : IF V_t is P and V_{t-1} is N and V_{t-2} is P and V_{t-3} is N THEN S is G

La detección uniforme de **intercambio de canales** presenta algunos problemas a la hora de establecer un sistema genérico. Por un lado, se puede pensar en controlar la distancia entre marcas a lo largo de la captura. En el momento en que la distancia entre cada par de marcas varíe respecto de su evolución histórica, estamos ante un intercambio de canales.

Esto exige la definición de una jerarquía de marcas o esqueleto interno, que nos permita realizar estas operaciones. Mediante la definición de dos sistemas (uno de detección del error y otro de identificación de parejas), se podrían reparar automáticamente este tipo de errores.

3.3 ANÁLISIS DE RESULTADOS

En la figura 5 se muestra el resultado de aplicar los sistemas de detección y corrección de errores sobre un intervalo de 70 frames de una captura real, en la marca correspondiente a la rodilla derecha de una persona tomando impulso para saltar. La trayectoria representada con trazos continuos muestra el resultado después de aplicar la corrección. Cada punto marcado en la gráfica se corresponde con la posición X, Y, Z de los canales asociados a cada marca en un instante de tiempo t . Como puede apreciarse, la trayectoria procesada se ajusta de forma correcta a la trayectoria original, produciendo un movimiento suave.

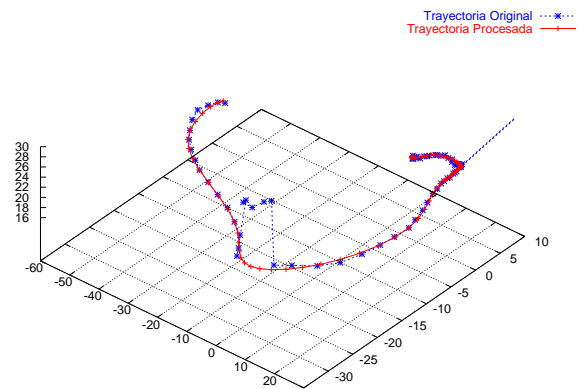


Figura 5: Resultado de aplicar los sistemas de detección y corrección de errores

El sistema de detección de marcas perdidas encuentra en el frame 27 un error puntual, debido a que la

velocidad en el canal respecto del eje X sufre un incremento muy alto. Esto puede observarse en la gráfica de la figura 6. En ese frame, se activa la regla R_2 y se aplica interpolación en el intervalo de frames 24-30.

En el intervalo de frames 47-53 hay una trayectoria errónea. Como puede verse en la figura 6, el sistema detecta el inicio del intervalo correctamente, y el final del mismo un frame después (debido al conjunto de reglas R_1 y R_2 en nuestro ejemplo). Esto no representa ningún problema, ya que la corrección se realiza tomando un intervalo de frames mayor. Debido a esto, la trayectoria reparada suaviza el camino seguido antes y después del intervalo erróneo.

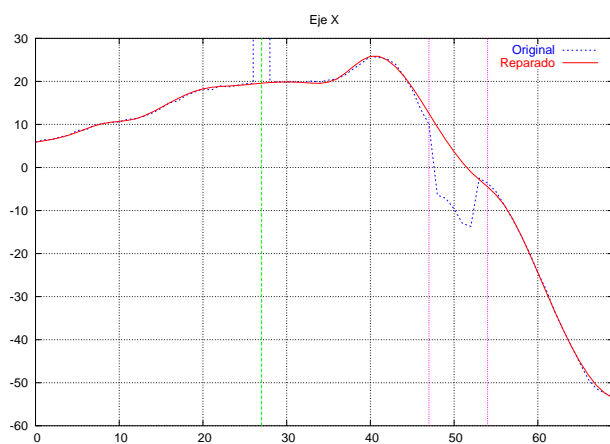


Figura 6: Resultado respecto del eje X

4 CONCLUSIONES

En este trabajo se ha presentado un método para la detección automática de errores en capturas de movimiento obtenidas mediante el uso de sistemas ópticos. La identificación de estos errores se ha realizado empleando conocimiento experto, que ha sido modelado utilizando las propiedades descriptivas de los sistemas de reglas basados en lógica difusa.

En futuros trabajos se avanzará en la incorporación de dependencias fuzzy entre articulaciones para el modelado de acciones complejas, definidas mediante un conjunto de sistemas dinámicos [4]. Esto nos permitirá detectar de forma sencilla las inconsistencias en los movimientos debidos al intercambio de canales, además de la generación cómoda de animaciones para personajes sintéticos y llegar a la definición puramente lingüística de acciones complejas.

Agradecimientos

Este trabajo ha sido financiado por los proyectos

PREDACOM PB-03-004 de la Junta de Comunidades de Castilla-La Mancha y DIMOCLUST TIC2003-08807-C02-02 del Ministerio de Ciencia y Tecnología.

Referencias

- [1] J. Foley, Computer Graphics: Principles and Practice, Addison Wesley, 1995.
- [2] C. Gonzalez-Morcillo, L. Jimenez, Mejora de un Sistema de Captura de Movimiento mediante modelos teóricos y algoritmos de Inteligencia Artificial: Proyecto Pin8. <http://www.infcr.uclm.es/www/cglez/pin8/Pin8Setup.exe>
- [3] E.H. Mamdani, S. Assilian, An experiment in linguistic synthesis with a fuzzy logic controller, *Int. J. Man-Mach, Studies*, 7, 1-13, 1975.
- [4] J. Moreno-Garcia, L. Jimenez, J.J. Castro-Schez, L. Rodriguez, A linguistic modelling approach for dynamic systems by using Temporal Fuzzy Models, *Subm. to Int. Journal of Approximate Reasoning*, 2004.
- [5] S.M. Netto, A. Evsukoff et al. Fuzzy Systems to Solve Inverse Kinematics Problem in Robots Control; Application to an Hexapod Robots' Leg. *Sixth Brazilian Symposium on Neural Networks*, 2000.
- [6] D. B. Olfe, Computer Graphics for Design. From Algorithms to AutoCAD. Prentice Hall, 1995.
- [7] K. Ogata. Ingeniería de control moderna. Prentice-Hall HispanoAmericana SA, 1998.
- [8] T. Takagi, M. Sugeno. Fuzzy identification of system and its application to modeling and control. *IEEE Transaction on system, Man and Cybernetics*, smc-15:116-132, January 1985.
- [9] K. Tanaka. An introduction to fuzzy logic for practical applications. Springer, 1998.
- [10] L. A. Zadeh. Fuzzy set. *Information and control*, 338-353, 1965.
- [11] L. A. Zadeh. The concept of a linguistic variable and its applications to Approximate reasoning. Part I, II, III. *Information Science*, 1975.