

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/228848310>

Integración de Múltiples Tareas Visuales en un Sistema Robótico

Conference Paper · July 2007

CITATIONS

0

READS

49

5 authors, including:



Daniel Hernández-Sosa

Universidad de Las Palmas de Gran Canaria

92 PUBLICATIONS 645 CITATIONS

[SEE PROFILE](#)



Jorge Cabrera

Universidad de Las Palmas de Gran Canaria

77 PUBLICATIONS 383 CITATIONS

[SEE PROFILE](#)



Antonio Carlos Domínguez Brito

Universidad de Las Palmas de Gran Canaria

63 PUBLICATIONS 338 CITATIONS

[SEE PROFILE](#)



Josep Isern

Universidad de Las Palmas de Gran Canaria

52 PUBLICATIONS 274 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



PLANCLIMAC (MAC2/3.5b/244): PLANIFICACIÓN CONJUNTA, SEGUIMIENTO Y OBSERVACIÓN, MEJORA DEL CONOCIMIENTO Y SENSIBILIZACIÓN ANTE RIESGOS Y AMENAZAS DEL CAMBIO CLIMÁTICO EN LA MACARONESIA [View project](#)



ICT COST Action IC1406: High-Performance Modelling and Simulation for Big Data Applications (cHiPSet) - Working Group 4 [View project](#)

Integración de Múltiples Tareas Visuales en un Sistema Robótico

Daniel Hernández
Jorge Cabrera
Angel Naranjo
Antonio Domínguez
Josep Isern

IUSIANI, ULPGC, España
dhernandez, jcabrera, adominguez@iusiani.ulpgc.es

RESUMEN¹

Los mecanismos de atención visual han sido frecuente objeto de interés por parte de investigadores en las áreas de robótica y visión por computador. Sin embargo, aspectos como el control de la mirada y la composición modular de capacidades visuales han sido estudiados en mucha menor medida. En este artículo se realiza una propuesta de arquitectura para sistemas robóticos móviles que incorporen visión activa. Esta propuesta ha sido concebida para facilitar el uso concurrente del sistema visual por parte de múltiples tareas cliente. Se describe en detalle la arquitectura funcional del sistema y se proporcionan soluciones para el problema del reparto de la mirada. El diseño del sistema oculta la complejidad de estos mecanismos de bajo nivel a los clientes, que pueden de este modo programarse como si fuesen usuarios en exclusiva del sistema visual. De forma complementaria, se contemplan principios de ingeniería del software para el diseño e integración, cuestiones que con excesiva frecuencia se abandonan en este tipo de desarrollos. En la parte final del artículo se presentan algunos resultados obtenidos sobre plataformas robóticas reales.

Palabras Clave: Visión por Computador, Control de la Mirada, Sistemas Robóticos Móviles.

1. INTRODUCCIÓN

El problema del control de la mirada en un sistema de visión activa (SVA) puede abordarse desde diferentes perspectivas. A menudo se formula como un problema de detección de puntos significativos en una imagen. Bajo este enfoque, diferentes aspectos como la relevancia

visual, el control bottom-up frente al top-down o el modelado computacional han sido analizados [1][2]. Una aproximación alternativa considera la naturaleza de recurso compartido del sensor visual, transformando el escenario en un problema de gestión/coordinación donde la mirada debe repartirse entre un conjunto dinámico de tareas concurrentes [3].

Consideramos que, además de los ya expuestos, existe un tercer escenario que merece la pena considerar. Desde un punto de vista más ingenieril, cabe destacar el hecho de que los investigadores e ingenieros involucrados en el desarrollo de los SVA se han concentrado principalmente en las capacidades visuales del sistema en términos de rendimiento, fiabilidad o integración del conocimiento. Sin embargo, se ha prestado escasa atención al problema de la composición modular de capacidades visuales en sistemas precepto-efectores, que siguen diseñándose e integrándose de una manera muy primitiva en base a los principios de la Ingeniería del Software moderna. Podemos establecer una simple analogía para clarificar este punto. Cuando ejecutamos programas que leen/escriben ficheros en el disco duro de un ordenador no tenemos que preocuparnos por problemas de contención relativos al uso de un recurso compartido. Los servicios de bajo nivel del Sistema Operativo se encargan de gestionar estos aspectos, simplificando la programación de estas aplicaciones que pueden ser desarrolladas como si tuvieran acceso exclusivo al dispositivo. En términos más generales, podríamos considerar esta característica como una “tecnología facilitadora” que permite la implementación de programas más complejos concentrando los esfuerzos en detalles de más alto nivel.

Cuántas de estas ideas se aplican actualmente al diseño de los SVA? En nuestra opinión, no demasiadas. Debido en gran medida a que esta área se ha considerado tradicionalmente como un reto investigador, con lo que los esfuerzos se derivan hacia otras cuestiones de más alto nivel, siendo estos detalles de bajo nivel normalmente ignorados. Si trasladamos el planteamiento expuesto anteriormente al contexto de los SVA, algunos

¹ Este trabajo ha sido financiado parcialmente por el Ministerio de Educación español, fondos FEDER (proyecto TIN2004-07087) y el Gobierno de Canarias (proyecto PI2003/160)

inconvenientes importantes se pueden identificar con facilidad:

- Los SVA tienden a ser sistemas monolíticos y cerrados, capaces de desarrollar un conjunto reducido de tareas predefinidas. Si se precisa la ejecución de múltiples tareas visuales concurrentemente hay que tenerlo presente desde la etapa de diseño, y los problemas de arbitraje de recursos se resuelven mediante código empotrado en las propias tareas implicadas.
- No existen mecanismos de integración que permitan incorporar de nuevas tareas que compitan por los recursos compartidos.
- Puesto que las situaciones de contención se gestionan internamente y son tratadas mediante soluciones ad-hoc, no se genera tecnología reutilizable en otros proyectos similares.

Como selección del trabajo relacionado, cabe destacar los siguientes sistemas. Dickmanns et al. [4] han estudiado el problema del control de la mirada en el contexto de su proyecto MarVEye, donde una cabeza activa multi-cámara es utilizada para conducir un vehículo en una autopista. En ese proyecto, diferentes regiones de interés en la imagen son propuestas y priorizadas por diferentes módulos de la arquitectura de control utilizando una medida de la ganancia de información. El controlador de la cabeza intenta planificar una trayectoria para la mirada que, dentro de una ventana temporal de 2 segundos, proporcione la mayor contribución de información para el sistema. Diferentes áreas de atención pueden encadenarse en una trayectoria, aunque en la práctica este número nunca es mayor que dos.

Los robots humanoides dependen de sus capacidades visuales para percibir el entorno. Incluso en las tareas de navegación simple, diferentes tareas deben operar concurrentemente en el sistema (evitación de obstáculos, localización, ...) para conseguir un mínimo resultado. Diferentes grupos de investigación han explorado el problema del arbitrio de la mirada en este escenario, tanto en simulaciones como con robots reales. Seara et al. [5][6] han experimentado con un robot bípedo que emplea una combinación de dos tareas para evitar obstáculos y localizarse mediante visión. La decisión relativa a dónde mirar se toma en dos fases. Primero, cada tarea selecciona su próximo foco de atención como aquél que le proporciona la mayor reducción de incertidumbre en la localización bien del robot, bien de los obstáculos. En una segunda fase, un esquema de decisión multiagente, en combinación con un modelo social de selección de ganador se utiliza para decidir a qué tarea se le otorga el control de la mirada. De los diferentes modelos de selección que se evalúan, los mejores resultados, a juicio de los autores, se obtienen con un modelo social que trata de minimizar la

“infelicidad colectiva”. Aquí, el concepto de infelicidad se deriva de perder la oportunidad de reducir la incertidumbre (en auto localización o en localización de obstáculos).

Sprague et al. [7][3] han diseñado un entorno de simulación donde un robot bípedo debe caminar sobre una línea trazada en el suelo mientras recoge basura y evita obstáculos, utilizando para ello únicamente la visión. Estas capacidades son implementadas como comportamientos visuales usando un método de aprendizaje por refuerzo para sintetizar la estrategia de control de la mirada óptima para cada tarea. La posición lateral del robot sobre la línea y la posición de los obstáculos y la basura se modelan mediante filtros de Kalman. Cada 300 milisegundos se cede el control de la mirada a la tarea que proporciona una mayor reducción en la incertidumbre.

Con una filosofía similar a la que subyace en los trabajos citados, la motivación de nuestra propuesta se expresa en una doble vertiente. Por un lado, queremos contribuir a la construcción de sistemas de visión más consistentes desde un punto de vista ingenieril, permitiendo la integración y composición modular de tareas en sistemas robustos que hagan un uso racional de los recursos disponibles. Por otro, se aspira a dar un primer paso hacia sistemas en los que la visión se integre en un contexto de acción con niveles semánticos y cognitivos más elevados (una forma “inteligente” de mirar).

En las siguientes secciones presentaremos la propuesta de arquitectura, con sus objetivos de diseño y componentes principales. Posteriormente analizaremos algunos resultados experimentales obtenidos sobre una plataforma robótica real. Para finalizar, se detallan las conclusiones y las líneas de trabajo futuro.

2. MTVS

Siguiendo las líneas expuestas en el apartado anterior hemos diseñado e implementado MTSV (Multi-Tasking Vision System), una propuesta de arquitectura para sistemas de visión activa en entornos multi-tarea. MTSV se ha diseñado para gestionar la planificación de la ejecución de tareas visuales concurrentes de manera que el arbitraje de los recursos permanece oculto al usuario.

2.1 Objetivos

Con un mayor nivel de detalle y concreción, MTSV persigue la consecución de los siguientes objetivos:

- El sistema debe ofrecer a los clientes un conjunto reducido de servicios o primitivas visuales en términos precategóricos.
- La asignación del control de la mirada a una tarea debe basarse en un modelo sencillo de planificador, fácilmente interpretable por un observador externo.

- Las tareas (clientes) se integran de manera individual y no coordinada en el sistema. El conjunto de tareas (actividad) que el sistema atiende puede cambiar dinámicamente.
- Los clientes no pueden asumir garantías en cuanto a tiempo de respuesta, aunque el sistema puede ofrecer modos de atención de alta prioridad.

2.2 Arquitectura interna

La figura 1 muestra una aplicación ejemplo en MTVS con dos clientes. Los elementos básicos que constituyen la arquitectura del sistema son los servidores principal y dedicados, los planificadores de tareas y el subsistema de adquisición de datos. Básicamente, los clientes se conectan con el servidor del sistema para solicitar servicios (primitivas visuales) con una determinada configuración. En respuesta, el servidor del sistema lanza dos hebras: un task-thread, para encargarse de los aspectos internos de la planificación de la ejecución, y un server-thread dedicado, responsable de la interacción con el cliente externo. El planificador analiza las demandas de las tareas y selecciona, en base a una determinada política, la que debe recibir el control de la mirada (en el ejemplo, el cliente A tiene el control). De manera simultánea, un planificador de atención encubierta (covert) determina el nivel de compatibilidad entre las peticiones de las tareas para ver si es posible compartir imágenes entre ellas (solapamiento de regiones de interés), incrementando de esta manera el rendimiento. El subsistema de adquisición de datos procesa las diferentes fuentes de datos (imágenes, pose de la cabeza, pose del robot) para generar, con la mayor precisión posible, etiquetas temporales y datos de pose para las imágenes servidas a los clientes.

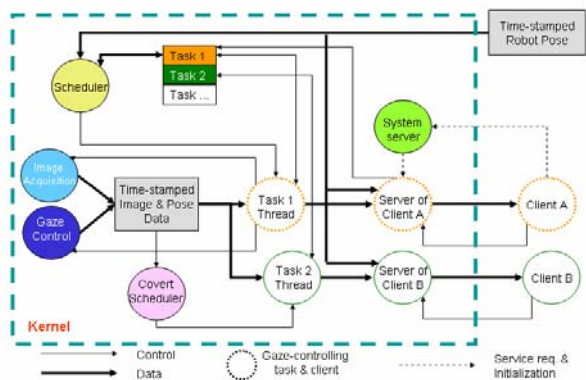


Figura 1 Arquitectura MTVS: ejemplo de aplicación con dos clientes.

2.3 Servicios Visuales

Los clientes interactúan con el sistema en términos de servicios visuales construidos a partir de un conjunto reducido de primitivas precategóricas de bajo nivel [8]:

- Obtener 1-N imágenes de un cierto punto del espacio y con una cierta configuración de cámara (WATCH).
- Obtener N imágenes a lo largo de una trayectoria (SCAN).
- Buscar un modelo de forma precategórica en una cierta zona (SEARCH).
- Seguir un modelo de forma precategórica (TRACK)
- Notificar cambios (movimiento, color, ...) (NOTIFY)

Salvo WATCH, el resto de las primitivas pueden ejecutarse de forma discontinua, permitiendo la implementación de tareas visuales interrumpibles.

Los clientes regulan su actividad en el sistema por medio de mensajes que intercambian con el servidor dedicado. Actualmente, los siguientes mensajes están disponibles: crear una nueva tarea (mensaje inicial), suspender una tarea, reconfigurar una tarea (modificar parámetros de cámara o de la tarea, su prioridad o urgencia, transformar una tarea primitiva en otra, etc), y aniquilar una tarea.

2.4 El Planificador

Se han implementado y estudiado diferentes políticas de planificación dentro de MTVS. Este análisis ha considerado dos grupos principales de planificadores: basados en el tiempo y basados en urgencia.

Se han estudiado tres tipos de planificadores temporales: Round-Robin (RR), Earliest Deadline First (EDF) y EDF con prioridades (EDFP). El algoritmo RR con prioridades reveló de inmediato su escasa utilidad en un esquema de acción dinámico y contextual. Primeramente, no tiene demasiado sentido asignar rodajas de tiempo similares a tareas diferentes; y en segundo lugar, el tiempo utilizado en movimientos sacádicos, especialmente cuando implican a un cuello lento, se desperdicia.

El algoritmo EDF ofreció un rendimiento ligeramente superior al RR, pero resultó difícil de generalizar dado que las tareas visuales no son adecuadas para su modelado como tareas periódicas. Los mejores resultados dentro de los planificadores temporales se obtuvieron con el algoritmo EDFP, combinando tareas críticas (deadline estricto) con tareas no críticas. Cada vez que una tarea es considerada para su ejecución y no resulta elegida se incrementa su prioridad en una cierta cantidad [9].

En lo referente a planificadores basados en urgencia cabe destacar que dicho concepto de urgencia está altamente correlacionado con un criterio de minimización de pérdidas, como consecuencia de que una cierta tarea no reciba el control de la mirada dentro de una determinada ventana temporal. Esta medida puede también ponerse en correspondencia con la incertidumbre en muchas tareas visuales.

Se implementaron dos planificadores en este segundo grupo: lotería [7] y urgencia máxima. El planificador basado en lotería emplea un esquema aleatorio de asignación de la mirada donde la probabilidad de que una tarea sea seleccionada es directamente proporcional a su urgencia (voto ponderado). De esta manera, se garantiza que todas las tareas tienen la posibilidad de adquirir el control, si bien esta falta de determinismo puede a veces causar efectos no deseados.

El planificador de máxima urgencia sustituye el mecanismo de voto ponderado por una selección directa de la tarea con máximo valor de urgencia. Este esquema ha producido resultados aceptables, siempre que se tenga en cuenta que la urgencia de una tarea debe reducirse significativamente después de adquirir el control de la mirada para evitar el bloqueo en exclusiva del recurso compartido (similar a un mecanismo de inhibición de retorno).

3. EXPERIMENTOS

Un conjunto de experimentos se han realizado al objeto de analizar el comportamiento de MTVS en una aplicación robótica real. Presentaremos en esta sección algunos de los resultados obtenidos.

El entorno experimental consiste en dos robots Pioneer de ActivMedia, uno con la configuración básica y otro sobre el que va montado el sistema de visión activa formado por una unidad PTU de Directed Perception (cuello) y una cámara motorizada EVI-G21 de Sony (ojo).

Dos tareas principales son combinadas en los diferentes experimentos: persecución de líder y evitación de obstáculos. La tarea de persecución comanda al robot con el SVA (perseguidor) para detectar y seguir un objetivo en forma de patrón rectangular especial montado sobre el segundo robot (líder), tratando de mantener entre ambos una distancia preestablecida. La tarea de evitación de obstáculos busca cilindros coloreados sobre el suelo estimando, con la mayor exactitud posible, su posición 2D. Se emplea filtrado de Kalman para modelar tanto la posición del objetivo como de los obstáculos.

3.1 Experimentos con una sola tarea

Como figura de mérito de referencia, en términos del máximo rendimiento esperable, se han realizado experimentos en los que sólo se ejecuta una de las tareas.

En el experimento de persecución aislado el robot líder se configura para moverse en línea recta a una velocidad constante de 200 mm/seg, mientras el perseguidor debe tratar de mantener una separación constante entre ambos de 2 metros. Se han realizado varias pruebas a lo largo de un pasillo de unos 15 metros de longitud. El robot perseguidor es capaz de estabilizar el error de seguimiento por debajo de los 150 mm.

En el experimento de evitación aislado el robot con el SVA es comandado para explorar el entorno en búsqueda de objetos (cilindros amarillos), tratando de reducir la incertidumbre relativa a su ubicación por debajo de un cierto umbral. El robot se mueve en línea recta dentro de un pasillo formado por 8 cilindros equiespaciados en zigzag. La figura 2 ilustra la trayectoria del robot y las diferentes detecciones para cada objeto localizado, incluyendo la primera (mayor) y la menor elipses de incertidumbre asociadas a cada una. Los resultados muestran como el robot es capaz de localizar la totalidad de los objetos con elipses de incertidumbre menores comprendidas entre 100 y 200 mm de diámetro.

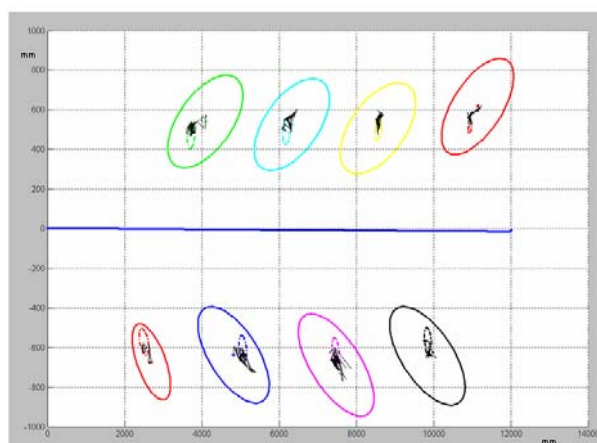


Figura 2 Tarea de evitación aislada.

3.2 Experimentos con múltiples tareas

En los experimentos con múltiples tareas se considera un escenario en el que cada cliente computa su urgencia y solicita al planificador MTVS el control de la mirada con una determinada configuración. El planificador utiliza esta información para seleccionar hacia dónde debe apuntarse la cámara y cómo distribuir las imágenes. La tarea de evitación de obstáculos se extiende para clasificar configuraciones especiales de objetos como “puertas” (dos objetos alineados perpendicularmente a la orientación inicial del robot con una separación preestablecida).

La urgencia de la tarea de seguimiento se computa como una función del error en la distancia de seguimiento, la velocidad del robot y el tiempo. Esta urgencia crece cuando la separación entre los robots se aleja del valor deseado, cuando la velocidad es alta o cuando el tiempo transcurrido desde que se recibió la última imagen es grande.

La urgencia de la tarea de evitación se computa separadamente para tres posibles focos de atención: zona frontal (la urgencia crece cuando el robot se mueve hacia áreas que no han sido exploradas visualmente), el objeto peor estimado (la urgencia se incrementa a medida que la posición de un objeto previamente detectado no es

confirmada con nuevas imágenes) y la puerta más cercana (urgencia mayor para puertas estrechas).

El primer experimento multitarea tiene el propósito de ilustrar el papel desempeñado por el planificador de atención encubierta de MTVS. En este caso una de las tareas tiene prioridad mayor, de manera que en un contexto de acceso exclusivo a las imágenes la tarea secundaria ofrece un rendimiento muy pobre. Permitiendo la compartición de imágenes, sin embargo, el rendimiento global puede mejorarse. El experimento final muestra un escenario más complejo, incluyendo detección de puertas.

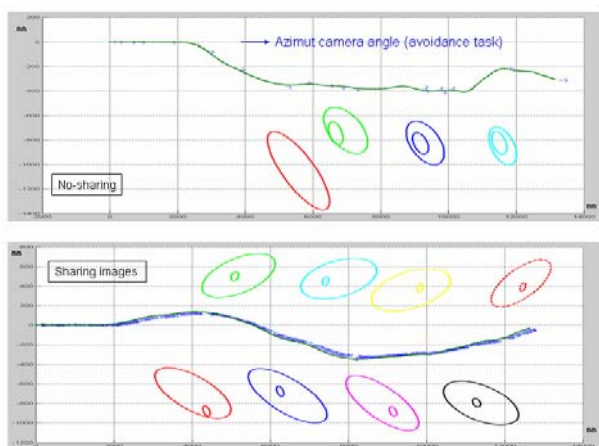


Figura 3 Evitación y seguimiento (prioritario).

- **Experimento de evitación y seguimiento compitiendo por el control de la mirada (prioridad para el seguimiento):** En este experimento el control de la mirada se cede a la tarea de evitación únicamente cuando tanto la velocidad del robot líder como el error en la distancia de seguimiento son bajos. Típicamente, el rendimiento de la tarea de seguimiento no se ve afectado significativamente por la competencia, al tener mayor prioridad. La tarea de evitación, sin embargo, se degrada resultando pocas detecciones de objetos con baja precisión. Como ejemplo, la gráfica superior de la figura 3 presenta los resultados de una ejecución con el planificador covert desactivado, donde sólo la mitad de los objetos potenciales han sido detectados (todos en la parte derecha debido a la posición del más cercano) con elipses de incertidumbre grandes. Como se muestra en la gráfica inferior, permitiendo la compartición de las imágenes se obtiene un mucho mejor rendimiento en la tarea de evitación. De nuevo se muestran las elipses primera y mínima para cada objeto.

- **Experimento de localización de puertas y seguimiento compitiendo por el control de la mirada:** La configuración de objetos utilizada en este experimento consiste en un conjunto de cuatro “puertas”: dos de tipo ancho (1500 mm) y dos de tipo estrecho (600 mm). Todas las puertas están localizadas en línea recta enfrente

del robot, la primera (ancha) a 3 metros y las siguientes cada metro y medio, alternando el tipo. El robot líder se comanda para moverse a velocidad constante, atravesando todas las puertas por el centro.

La figura 4 ilustra cómo la cámara apunta a ambos extremos cuando cruza puertas estrechas. Como consecuencia de este comportamiento, el robot seguidor ralentiza su desplazamiento al aproximarse a una puerta estrecha hasta que su posición ha podido determinarse con suficiente precisión (puede compararse el tamaño final de las elipses en las puertas de cada tipo). Tras cruzar la puerta, el robot acelera para tratar de recuperar la distancia de seguimiento deseada con respecto al líder.

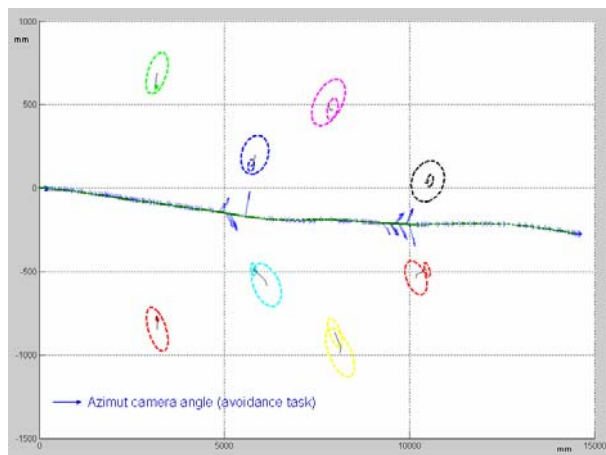


Figura 4 Seguimiento y localización de puertas.

4. CONCLUSIONES Y TRABAJO FUTURO

En este artículo, proponemos una arquitectura abierta para la integración de tareas visuales concurrentes en sistemas de visión activa. Las peticiones de los clientes se articulan sobre la base de un conjunto reducido de servicios o primitivas visuales. Todos los aspectos de bajo nivel ligados al control/coordinación se ocultan a los clientes, simplificando su programación y permitiendo la composición dinámica de la actividad visual desde capacidades mucho más básicas.

En referencia al problema de la asignación del control de la mirada, diferentes planificadores han sido implementados y analizados. Los mejores resultados hasta el momento se han obtenido por un esquema contextual gobernado por urgencias, tomando la interacción del agente con su entorno y no simples frecuencias temporales como principio de organización. Normalmente, puede establecerse una correspondencia entre la urgencia y la incertidumbre relativa a algún elemento de la tarea.

El sistema descrito en este artículo es un simple prototipo, principalmente un demostrador de conceptos, y

es susceptible de mejora en múltiples facetas. Así, por ejemplo, tenemos previsto mejorar la adaptabilidad del sistema a diferentes cabezas de visión activa (abstracción hardware). Un primer paso es considerar una extensión del sistema para cabezas binoculares, donde nuevos problemas como coordinación ocular, vergencia y acomodación deben analizarse. Otra mejora a introducir es la inclusión de un test de aceptación de nuevas tareas, a fin de evitar la sobrecarga del sistema. Adicionalmente, la incorporación de mecanismos de control homeostático contribuirán a hacer al sistema más robusto; y la atención bottom-up permitiría ampliar el repertorio de primitivas de relevancia visual disponibles (detección de movimiento independiente, por ejemplo).

5. REFERENCIAS

- [1] Arkin, R., ed.: *Behavior-Based Robotics*. MIT Press (1998).
- [2] Itti, L.: *Models of bottom-up attention and saliency*. In Itti, L., Rees, G., Tsotsos, J.K., eds.: *Neurobiology of Attention*. Elsevier Academic Press (2005).
- [3] Sprague, N., Ballard, D., Robinson, A.: *Modeling attention with embodied visual behaviors*. *ACM Transactions on Applied Perception* (2005).
- [4] Pellkoffer, M., Ltzeler, M., Dickmanns, E.: *Interaction of perception and gaze control in autonomous vehicles*. In: *SPIE: Intelligent Robots and Computer Vision XX: Algorithms, Techniques and Active Vision*, Newton, USA (2001).
- [5] F. Seara, J., Lorch, O., Schmidt, G.: *Gaze Control for Goal-Oriented Humanoid Walking*. In: *Proceedings of the IEEE/RAS Humanoids, Tokio, Japan (November 2001)* 187–195.
- [6] F. Seara, J., Strobl, K.H., Martin, E., Schmidt, G.: *Task-oriented and Situationdependent Gaze Control for Vision Guided Autonomous Walking*. In: *Proceedings of the IEEE/RAS Humanoids, Munich and Karlsruhe, Germany (October 2003)*.
- [7] Sprague, N., Ballard, D.: *Eye movements for reward maximization*. In: *Advances in Neural Information Processing Systems (Vol. 16)*. MIT-Press (2003).
- [8] Christensen, H., Granum, E.: *Control of perception*. In Crowley, J., Christensen, H., eds.: *Vision as Process*. Springer-Verlag (1995)
- [9] Kushleyeva, Y., Salvucci, D.D., Lee, F.J.: *Deciding when to switch tasks in timecritical multitasking*. *Cognitive Systems Research* (2005).