

Distribución de una Red de Cámaras usando Algoritmos de Búsqueda Codiciosa y Genéticos

Martha L. Quiñones-Muñoz¹, Elias Carrum-Siller¹, L. Abril Torres-Méndez² y Pedro Pérez-Villanueva¹

¹Corporación Mexicana de Investigación en Materiales S.A. de C.V.
Ciencia y Tecnología No. 790, Col. Saltillo 400, C.P. 25290 Saltillo, Coahuila, México.

²Grupo de Robótica y Manufactura Avanzada, Cinvestav Saltillo.
Carretera Saltillo-Monterrey Km. 13.5, Saltillo, Coahuila, México.

mquinones@comimsa.com

Resumen. En este artículo se aborda el problema de la distribución de un conjunto de cámaras en un espacio de trabajo cerrado, con el objetivo de visualizar un área de interés en su totalidad en cada instante de tiempo. La metodología a seguir en este proyecto se basa en modelos de geometría computacional y proyectiva, así como características de detección y algoritmos de búsqueda. La experimentación se realiza empleando la misma metodología, variando únicamente los algoritmos de búsqueda que se implementarán, siendo estos algoritmos genéticos y búsqueda codiciosa (*greedy search* en inglés), con la finalidad de realizar una comparación de los resultados obtenidos con cada uno de los algoritmos mencionados, evaluando la cobertura del área de interés obtenida, así como el tiempo de cómputo involucrado en el análisis.

Palabras clave: Red de cámaras, Algoritmos Genéticos, Greedy search.

1 Introducción

En la actualidad el uso de redes de cámaras en diversas actividades se ha vuelto imprescindible, desde aplicaciones robóticas que dependen de información visual para realizar una o varias tareas, hasta en sistemas de vigilancia utilizados en locales comerciales, aeropuertos o sistemas de monitoreo de tráfico. La mayoría de los sistemas o redes de cámaras solo capturan y transmiten los datos visuales a una computadora central, para ser analizados por operadores humanos y la única tarea automatizada involucra el procesamiento de imágenes en cámaras individuales.

A medida que aumenta el número de cámaras, se torna imposible para el ser humano el análisis efectivo de grandes cantidades de información dentro de las secuencias de video. Por tal motivo, la distribución de una red de cámaras en el espacio de trabajo es de gran relevancia, ya que si la distribución de la red es adecuada es posible emplear la mínima cantidad de cámaras, facilitando así el análisis de la información adquirida. Sin embargo, en muchas

de las ocasiones esto no es posible, debido a que no se hace un análisis a fondo y sólo se cuenta con la experiencia humana, con lo cual no se asegura la visualización de las áreas de interés en su totalidad con la menor cantidad de cámaras. Es por esto que en el presente trabajo se comparan dos técnicas heurísticas, búsqueda codiciosa y algoritmos genéticos, que brindaran la posición y orientación con la cual se visualiza una determinada área de interés.

La distribución de un conjunto de cámaras se puede hacer manualmente, sin embargo, esto está sujeto a errores tal como la incertidumbre humana, entre otros factores, lo cual ha motivado a varios investigadores a realizar análisis minuciosos al respecto, por lo que en la literatura existen varios métodos automatizados para encontrar una distribución óptima para redes de cámaras dentro de un espacio de trabajo determinado. Dichos métodos se basan principalmente en dos técnicas para estimar la posición y orientación óptima de las cámaras; la primera consiste en utilizar información visual proveniente de las

cámaras para realizar su análisis [1-2], esto es mediante el rastreo de ciertas características visuales de un objeto en movimiento dentro del espacio a supervisar, e inferir con ello la trayectoria del objeto; la información obtenida es utilizada para identificar y correlacionar probabilísticamente el objeto en movimiento en diferentes puntos de vista de las cámaras.

La segunda técnica se basa en la geometría proyectiva de las cámaras, empleando la proyección del campo de vista de las cámaras para cubrir los puntos característicos estáticos dentro del espacio de trabajo [3-10], los cuales determinan el área de interés a supervisar por la red de cámaras, esto con la finalidad de que los puntos sean observados por el campo de vista de cada una de las cámaras, encontrando así la distribución óptima de las cámaras dentro de un espacio de trabajo. El presente trabajo se enfoca en la implantación de esta técnica.

Dentro de las investigaciones realizadas usando la técnica de geometría proyectiva se encuentran algunos trabajos que se han enfocado a comprender la topología de una

red de cámaras cuyos campos de vista no se traslapen [3-6]. La idea principal está en comprender el trayecto o áreas que se desean visualizar e integrarlas en una configuración tal que la información visual pueda ser capturada desde posiciones estratégicas de las cámaras. Típicamente, los métodos para la distribución de un conjunto de sensores requieren información de distancia, conocida como rango del sensor, la cual, en el caso de las cámaras representa la distancia proyectiva a la que los objetos de la escena a visualizar se perciben de manera clara y nítida. De esta forma, una vez definidos los puntos característicos que describen las regiones de interés en la escena, se puede resolver el problema de ubicación óptima de las cámaras empleando una malla de puntos [7-8], dicho enfoque presenta variaciones de acuerdo al método de programación lineal utilizado. Sin embargo, es importante notar que estos métodos obtienen resultados favorables sólo cuando el problema a tratar considera un bajo número de puntos en la malla, ya que al aumentar el número de variables del problema, estos se

convierten en problemas del tipo NP duro, el cual incrementa el costo computacional.

Así mismo, se encuentran en la literatura diversos artículos que resuelven el problema de la distribución de una red de cámaras, empleando métodos heurísticos [9-10], como son la búsqueda codiciosa y algoritmos genéticos, los cuales muestran resultados favorables en comparación a los de programación lineal. Por tanto, el objetivo de este artículo es realizar una implementación y comparación de los algoritmos de búsqueda codiciosa y genético, enfocados a la resolución del problema de cobertura, con la finalidad de compararlos y deducir para qué casos brindan mejores resultados cada uno de ellos. A continuación, se describirá la metodología usada para ambos algoritmos, una vez realizada esta explicación se mostrarán los resultados obtenidos de dicha comparación.

2 Metodología

En aplicaciones de vigilancia interior donde la información visual es fundamental, por lo general se requiere

de dos o más cámaras para visualizar ya sea un área grande o varias regiones pequeñas de interés. La tarea de distribuir de manera óptima las cámaras para garantizar la visualización completa de dichas regiones en cada momento, es un problema conocido como cobertura óptima [9,11].

El problema de cobertura se puede definir como: dado un conjunto de n cámaras y un conjunto de m tareas para llevarse a cabo en un espacio de configuración cercano; se desea generar una solución que cubra todo el espacio de configuración de una manera eficiente, es decir, aprovechando al máximo el área del campo de vista de la cámara donde esta obtiene una mejor visualización (eje óptico, distancia proyectiva). En seguida, se muestra el modelo matemático que se resuelve en este trabajo.

$$\max \sum_{i=1}^n CP_i$$

Sujeto a:

$$si \begin{cases} existe\ obstaculo = 0 \\ de\ lo\ contrario\ CP_i \end{cases} \begin{cases} \|d_{ij}\| \leq r_i \\ \delta_{ij} < \frac{\phi_i}{2} \end{cases},$$

donde:

$$0 \leq \|d_{ij}\| \leq 2$$

$$0 \leq \delta_{ij} \leq 2$$

CP_i

= Puntos de control cubiertos por la cámara i

n = Num. total de cámaras

d_{ij}

= Vector unitario que va de la cámara i al punto de control P_j

δ_{ij} = Ángulo entre el eje óptico y d_{ij} .

ϕ_i = FOV de la cámara i

En el cual i representan la cámara y j el punto de control que están siendo analizados.

A continuación, se describe la metodología que lleva a cabo cada uno de los algoritmos a fin de resolver el modelo matemático planteado.

2.1 Algoritmo búsqueda codiciosa

El algoritmo de búsqueda codiciosa (Greedy search) es un método heurístico que se basa en escoger la mejor opción en cada iteración. Para la implementación de este algoritmo

se basó en el artículo presentado por Horster y Lienhart [9].

Parámetros y conjuntos

Para el análisis de la ubicación de una red de cámaras se emplea el conjunto de los puntos de control Ω , el cual está representado por las coordenadas del área de interés que se desea visualizar.

El conjunto de posibles orientaciones D_i se define para cada una de las cámaras disponibles. Este conjunto, se calcula de manera discreta y por intervalos, tomando valores entre 0° y 360° .

Por último se definen el conjunto de las posibles posiciones que pueden tomar las cámaras (Π), para esto se utiliza la envolvente convexa [12] (en inglés convex hull), uno de los constructores geométricos fundamentales. Este constructor, como su nombre lo indica, envuelve una nube de puntos en dos o más dimensiones dentro de un polígono convexo. El conjunto Π está formado por los puntos que se encuentran dentro del área delimitada por la envolvente convexa, la cual está conformada por los puntos que se

desean visualizar (Ω) y por las aristas de los obstáculos presentes en el área de trabajo, delimitando así el área de búsqueda dentro del espacio de trabajo.

Algoritmo

El algoritmo de cobertura inicia con el ordenamiento de manera descendente de las cámaras disponibles con respecto a su campo de vista (ϕ_i). Una vez ordenadas las cámaras se selecciona la de mayor campo de vista (FOV) y se establecen los parámetros de entrada correspondientes a esta cámara, los cuales están representados por las posibles orientaciones, los puntos de control, posibles posiciones así como las características de las cámaras y el ambiente.

El análisis comienza descartando la existencia de colisión entre el eje óptico de la cámara C_i , originado a partir de la combinación de las posibles posiciones Π y orientaciones D_i que puede adquirir la cámara, y los obstáculos presentes en el ambiente, esto es debido a que si se interceptan se desperdicia el área de cobertura de la cámara. Al

no existir colisión, significa que la posición y orientación dada es candidata para representar a la configuración que resolverá el problema, en caso de existir colisión se descarta esa configuración y se prueba con otra hasta obtener la configuración con la cual no exista colisión y poder así continuar con el análisis.

La segunda etapa del análisis consiste en la formación de la matriz de cobertura θ_{ij}^d , la cual contiene los valores que indican la cobertura de los puntos de control $P_j \in \Omega$ que tiene una cámara C_i en cada una de las configuraciones candidatas. Para que un punto de control P_j sea cubierto por el campo de vista de una cámara C_i se deben de cumplir las siguientes dos condiciones (en la figura 1 se muestra gráficamente los términos involucrados en estas condiciones):

- $\|d_{ij}\| \leq r_i$, donde d_{ij} corresponde al vector unitario que va de la cámara al punto de control P_j y r_i es el rango de vista o distancia proyectiva de la cámara.
- $\delta_{ij} < \frac{\phi_i}{2}$, donde δ_{ij} corresponde al ángulo entre v_d^i (eje óptico) y d_{ij} .

-

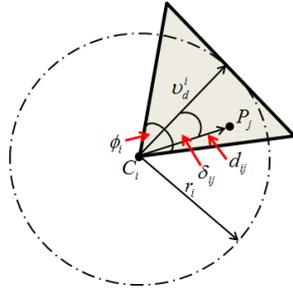


Fig. 1. Términos que determinan la cobertura de una cámara.

La matriz de cobertura obtiene valores entre cero y dos, debido a que se asigna una ponderación dependiendo de la ubicación que tenga un punto de control P_j con respecto a la configuración de la cámara y su campo de vista, en otras palabras, dependiendo de los valores de las variables d_{ij} y δ_{ij} de las condiciones de cobertura establecidas anteriormente.

Después de analizar las combinaciones posibles entre los elementos de los conjuntos Π y D_i , se elige la configuración $i - d$ (posición y orientación) con la que se cubra la mayor cantidad de puntos de control, de acuerdo a la ponderación descrita anteriormente.

Una vez seleccionada la posición y orientación para la cámara C_i , que cubra la mayor cantidad de puntos de control, se eliminan los puntos de control cubiertos del conjunto Ω , siendo este el último paso de la iteración y la base para la siguiente iteración del análisis.

Con este algoritmo es posible brindar una solución al problema planteado en este trabajo, solución que dependerá del tamaño de la muestra de los conjuntos Π , D_i y Ω . El caso ideal es cuando estos conjuntos son continuos, pero debido al costo computacional que conllevaría realizar este análisis es que en este trabajo se considera a estos conjuntos como discretos.

2.2 Algoritmo genético

El algoritmo genético (por sus siglas en inglés GA) es el algoritmo evolutivo más usado. GA fue desarrollado por Holland en 1975 [13] y ha crecido como el paradigma más utilizado para resolver problemas de optimización [14]. Hay varias variantes de los algoritmos genéticos, sin embargo, todos tienen cuatro procedimientos generales: la evaluación de los

individuos, donde se obtiene el resultado de la función objetivo, seguido a esto se realiza una selección de los mejores individuos, la cual puede ser elitista, torneo, ruleta, entre otras. Una vez seleccionados los mejores individuos se realiza el proceso de variación, el cual consiste en el cruce y la mutación de los individuos [15]. Cada algoritmo contiene una población de individuos, los cuales representan un conjunto de soluciones, estos individuos pueden ser simbolizados ya sea como un vector binario, de números reales, entre otros [14], mismos que hacen una analogía de los procesos naturales.

El genotipo de los individuos que se emplean para este análisis es un vector binario conformado por tres parámetros por cámara involucrada en el estudio, los cuales representan la orientación, la posición respecto al eje x y la posición respecto al eje y de cada una de las cámaras (Fig. 2.).

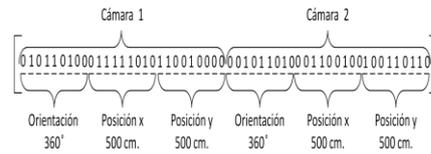


Fig. 2. Ejemplo de genotipo de 54 bits, para el análisis de la ubicación de dos cámaras dentro de una habitación de 5m. x 5m.

En el presente trabajo se empleó un algoritmo genético elitista no generacional [13], para encontrar la configuración de una red de cámaras con la cual se pueda visualizar cierta área de interés, a continuación se presentan los pasos que se siguieron para implementar este algoritmo.

El primer paso consiste en la inicialización de la población, la cual se realiza de manera aleatoria, con 100 individuos como población.

En el segundo paso se lleva a cabo la evaluación de la población, en donde se evalúan cada uno de los individuos y se le asigna una ponderación de acuerdo a la solución que brinda al problema. Esta etapa emplea las mismas condiciones que la matriz de cobertura del

algoritmo de búsqueda codiciosa, para evaluar la configuración que

Donde:

Ancho = Ancho de la habitación.	O= Obstáculo.
CP = Puntos de control.	P = Población.
DP = Distancia proyectiva.	PC = Probabilidad de cruce.
FOV = Campo de vista.	Pdec = Población descodificada.
Largo = Largo de la habitación.	PE = Población evaluada.
Ncam = Número de cámaras.	PM = Probabilidad de mutación.
NG = Número total de generaciones.	PS = Población seleccionada.
NTI = Número total de individuos.	VMP = Valor máximo de la población.
NTB = Número total de bits (tamaño del genotipo).	

brinda cada uno de los individuos de la población.

En el tercer paso se emplea el

método de selección de torneo para elegir a los individuos más aptos de la población, es decir a la configuración que cubre la mayor cantidad de puntos de control.

El cuarto paso es la generación de los nuevos individuos utilizando la población seleccionada, a esta se le aplica el proceso de cruce, el cual consiste en el cruce de un punto, además se le aplica el proceso de mutación el cual es realizado mediante la mutación por intercambio.

Como se mencionó anteriormente el algoritmo genético implementado en este trabajo es un algoritmo elitista no generacional, por lo cual el mejor individuo a lo largo de las distintas generaciones se mantiene intacto hasta que existe un individuo mejor que lo reemplaza.

A continuación se muestra el pseudocódigo del GA implementado en este trabajo.

[P]←generar población inicial (NTI,NTB)

Principio de ciclo 1: NG

[Pdec]←decodificación (P,NTI)

[PE]←evaluación(Pdec,CP,NTI,Ncam,O,DP,FOV,Largo,Ancho)

[VMP]← localiza el máximo (PE)

[PS]←selección de torneo
(PE,P,NTI)
[P]←cruce y mutación
(PS,NTI,PC,PM,NTB)
Fin de ciclo

En la parte experimental del algoritmo genético se emplearon los siguientes parámetros, los cuales fueron obtenidos a través de varias pruebas, observando un mejor comportamiento del algoritmo con estos parámetros:

- Número de individuos =100
- Probabilidad de mutación = 0.15
- Número de generaciones =300
- Probabilidad de cruce= 0.85
-

3 Resultados

En esta sección se presentan los resultados obtenidos tras la implementación del algoritmo de búsqueda codiciosa y el algoritmo genético, empleados para obtener la configuración de una red de cámaras con la cual se pueda visualizar cierta área

de interés de una manera eficiente (aprovechando la región de la cámara donde esta obtiene una mejor visualización del objeto de interés).

Los resultados obtenidos se presentan en dos casos, los cuales son instalaciones generadas de manera aleatoria, variando la dimensión de las habitaciones, así como la cantidad de puntos de control a visualizar; con lo cual se muestra el funcionamiento y la capacidad de cada uno de los algoritmos para resolver el problema de cobertura.

Para la obtención de los resultados del algoritmo genético que se muestran a continuación se realizaron 35 pruebas, de las cuales se muestra la media de ellas para cada uno de los casos.

Caso 1.

El primer caso se lleva a cabo en una habitación de 5 metros de largo por 5 metros de ancho, con un obstáculo presente. Se consideran tres cámaras iguales para el análisis, las cuales tienen un campo de vista de 90° y una distancia proyectiva de 3m.

La trayectoria que se desea visualizar está compuesta por 86 puntos de control (CP), y se determinan los valores de dichos punto, de acuerdo a su posición dentro del campo de vista de la cámara.

En la tabla 1 se comparan los resultados obtenidos por ambos algoritmos, en la cual se puede apreciar como en los dos casos cumplen con el objetivo de cubrir el área de interés en su totalidad, sin embargo el algoritmo de búsqueda codiciosa no cubre los puntos de control de la mejor manera, ya que tiene 108.7433 en el valor que se le asignan a estos de acuerdo a su posición dentro del campo de vista de las cámaras; a diferencia del GA, el cual cubre mejor los puntos de control, pues su valor en la misma categoría es de 128.0663, con lo que se aprovecha efectivamente el FOV de las cámaras, donde se puede visualizar mejor un objeto (en las figuras 3 y 4 se visualiza gráficamente ambas soluciones).

Respecto al tiempo de cómputo se muestra una desventaja considerable del GA, ya que consume 3.8 veces más de tiempo que el algoritmo de búsqueda codiciosa.

Tabla 1. Resultados obtenidos en el primer caso.

	Búsqueda codiciosa	GA
Tiempo de computo (Segundos)	426.7719	1638.436005
Cantidad de CP cubiertos	86	86
Valor de los CP cubiertos de acuerdo a su posición	108.7433	128.0663

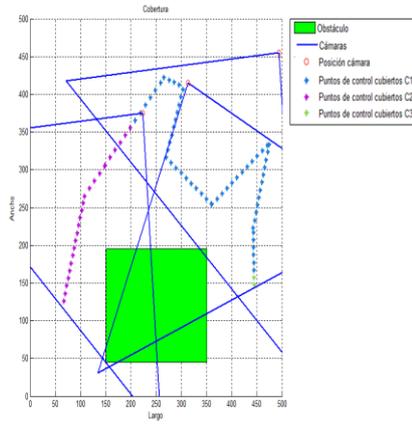


Fig. 3. Distribución de una red de cámaras obtenida por el algoritmo de búsqueda codiciosa para el caso 1.

1.

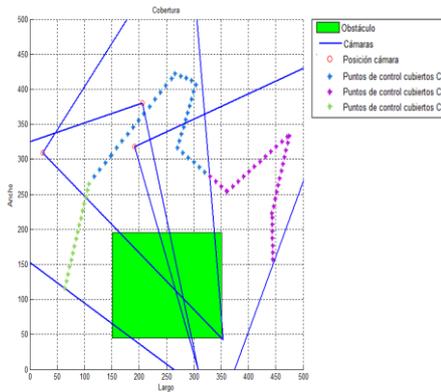


Fig. 4. Distribución de una red de cámaras obtenida por el algoritmo genético para el caso 1.

Caso 2.

El segundo análisis se efectuó en un espacio de trabajo cerrado de 10 metros de largo por 7 metros de ancho, en el que se encuentran 2 obstáculos presentes. Para el análisis se consideran 3 cámaras iguales, las cuales poseen un campo de vista de 90° y una distancia proyectiva de 3.5 metros. El área de interés que se desea visualizar es de 294 puntos de control.

Tabla 2. Resultados obtenidos en el segundo caso.

	Búsqueda codiciosa	GA
Tiempo de computo (Segundos)	6049.6	17520
Cantidad de CP cubiertos	234	236
Valor de los CP cubiertos de acuerdo a su posición	287.0582	288.2170

En la tabla 2, se muestra comparativamente los resultados obteni-

dos con cada uno de los algoritmos. En donde al igual que en el caso anterior se puede observar que el GA brinda un mejor resultado respecto al valor de los puntos de control que son cubiertos de acuerdo a su posición dentro del FOV. Además para este caso el GA cubre una mayor cantidad de puntos de control.

En las figuras 5 y 6, se muestra gráficamente los resultados obtenidos con el algoritmo de búsqueda codiciosa y el algoritmo genético, respectivamente; donde se puede apreciar gráficamente la posición y orientación con la que solucionan el problema, teniendo únicamente tres cámaras disponibles, con lo que se puede observar que únicamente con tres cámaras no se puede dar solución al problema, ya que el espacio de trabajo es muy grande y se tienen demasiados puntos de control, por lo que se requiere de una cámara extra para cumplir con el objetivo de visualizar el área de interés en su totalidad.

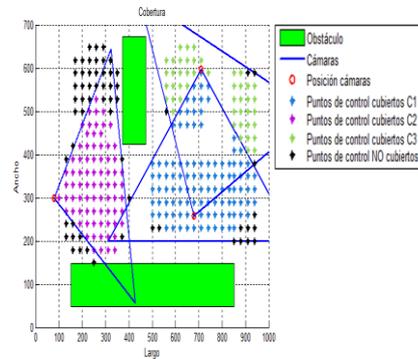


Fig. 5. Distribución de una red de cámaras obtenida por el algoritmo de búsqueda codiciosa para el caso 2.

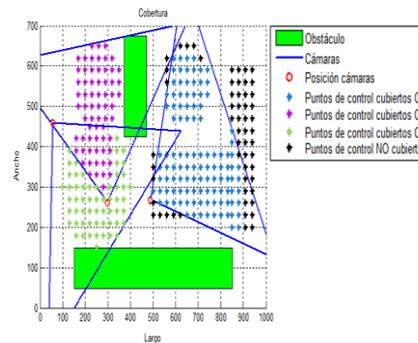


Fig. 6. Distribución de una red de cámaras obtenida por el algoritmo genético para el caso 2.

El comportamiento de la evolución de la población empleada en el GA para cada uno de los casos (1 y 2) se muestra en la figura 7, en la cual se puede apreciar como en

cada generación el algoritmo brinda una mejor solución al problema, sin embargo, el algoritmo no puede dar como resultado el valor máximo que puede tomar la matriz θ_{ij}^d (172 y 588, para el caso 1 y 2 respectivamente), a pesar de que en el primer caso se cubren todos los puntos de control que se desean visualizar, esto es debido a la ponderación que se le da al punto de control dentro del campo de vista de la cámara.

4 Conclusiones y trabajo futuro

Este artículo se enfoca en la implementación y la comparación de dos algoritmos heurísticos, búsqueda codiciosa y algoritmo genético; los cuales resuelven el problema de la distribución de un conjunto de cámaras en un ambiente de interior cerrado, tal que estas sean capaces de monitorear ciertas regiones de interés.

Con los resultados obtenidos se puede llegar a la conclusión de que el algoritmo que brinda la mejor solución al problema, considerando la cantidad de puntos de control y su valor respecto a su posición dentro de campo de vista de las cámaras, es el algoritmo genético, ya que este aprovecha más las áreas de la cámara donde estas visualiza mejor a los objetos (sobre el eje óptico y a la distancia proyectiva de la cámara), centrando los puntos de control respecto a el eje óptico de la cámara.

La única desventaja del AG respecto al algoritmo de búsqueda codiciosa es el tiempo de cómputo ya que este consume aproximadamente 3 veces más de tiempo.

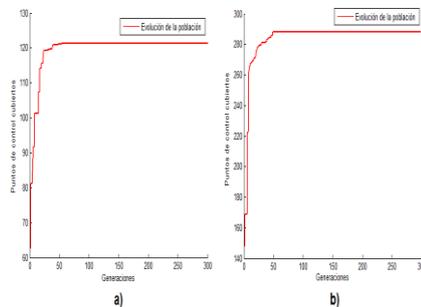


Fig. 7. Comportamiento de la evolución de la población empleada en el GA para el caso 1 (a) y el caso 2 (b).

Como expectativas de trabajo futuro, se propone llevar a la práctica la implementación de ambos algoritmos, analizar los resultados obtenidos en un caso real y de manera física, de manera que se pueda comprobar de modo práctico el desempeño que brinda cada uno de los algoritmos.

A su vez, se propone extender el análisis a cámaras dinámicas, es decir, cámaras que tendrían un movimiento individual controlado por las características visuales del ambiente. Al ser dinámicas, se reduciría la cantidad de cámaras necesarias para visualizar eficientemente el área de interés en cada instante de tiempo.

Agradecimientos

Los Autores agradecen el apoyo recibido para la investigación al CONACYT-FORDECYT proyecto 2143332.

Referencias

1. C. Taylor, A. Rahimi, J. Bachrach, H. Shrobe, A. Grue. (2006). "Simultaneous localization, calibration, and tracking in an ad hoc sensor network". Proceedings of the 5th International Conference on Information Processing in Sensor Networks, 27-33.
2. S. Funiak, C. Guestrin, M. Paskin, R. Sukthankar. (2006). "Distributed localization of networked cameras". Proceedings of the 5th International Conference on Information Processing in Sensor Networks, 34-42.
3. Makris D., Ellis T. and Black J. (2004). "Bridging the gaps between cameras". Computer Vision and Pattern Recognition Vol. 2, 205–210.
4. Marinakis D., Dudek G. and Fleet D. (2005). "Learning sensor network topology through Monte Carlo Expectation Maximization". International Conference on Robotics and Automation, 4581–4587.
5. Niu C. and Grimson E. (2006). "Recovering non-overlapping network topology using farfield vehicle tracking data". International Conference on Pattern Recognition Vol. 4, 944–949.
6. Tieu K., Dalley G. and Grimson E. (2005). "Inference of nonover-

- lapping camera network topology by measuring statistical dependence". International Conference on Computer Vision pp. 1842–1849.
7. Chakrabarty K, Iyengar S S, Hairong and Cho E (2002). "Grid Coverage for Surveillance and Target Location in Distributed Sensor Networks". IEEE Transactions on Computers pp. 1448 – 1453.
 8. Horster E and Lienhart R (2006). "Calibrating and optimizing poses of visual sensors in distributed platforms". Multimedia Systems 12, 195–210.
 9. Horster E and Lienhart R (2009). "Multi-Camera Networks: Concepts and Applications". Elsevier, ISBN-13: 978-0-12-374633-7.
 10. Pierre David, Vincente Idasiak and Frédéric Kratz. (2007). "A sensor placement approach for monitoring of indoor scenes". European Conference on Smart Sensing and Context, LNCS, 110-125.
 11. Osais Y, St-Hilaire M and Yu. F R (2009). "On Sensor Placement for Directional Wireless Sensor Networks". IEEE International Conference on Communications pp. 1–5.
 12. De Berg M., Cheong O., Van Kreveld M. and Overmars M. (2008). "Computational Geometry: Algorithms and Applications". Springer, 3rd. Edition.
 13. Holland, J. H. (1975). "Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence". University of Michigan Press, Ann Arbor.
 14. Mitsuo, Gen (2000). "Genetic algorithms & engineering optimization", John Wiley & Son, Chapter 9.
 15. David E. Goldberg (1996). "Genetic algorithms in search, optimization, and machine learning". Oxford University Press, New York.



Martha L. Quiñones-Muñoz es Ingeniero Industrial por el Instituto Tecnológico de Delicias y Maestra en Ciencias en Robótica y Manufactura Avanzada por el Centro de Investigación y Estudios Avanzados del Instituto

Politécnico Nacional, unidad Saltillo, Coahuila. Sus líneas de investigación comprenden visión por computadora e inteligencia artificial. Actualmente se encuentra laborando como investigadora en la celda de manufactura avanzada y en el departamento de Ingeniería Industrial en la Corporación Mexicana de Investigación en Materiales S.A. de C.V.



Elias Carrum-Siller es Ingeniero industrial y de sistemas por parte de la universidad autónoma del noreste y maestro en ciencias y tecnología y doctorado en ciencia y tecnología con especialización en ingeniería industrial y de sistemas de manufactura por parte del posgrado interinstitucional en ciencia y tecnología. Del año 1999-2001 se desempeñó en el área de Ingeniería de Proyectos para el Grupo Electromecánico Coahuilense, donde se encargó de la creación de carros de abastecimiento Poke-Yoke, monitoreo y creación de números Kan-Ban para la fabricación de piezas y la introducción de la filosofía de manufactura esbelta en el proceso de fabricación. De 2001-

2002 se desempeñó en General Electric EDESA en la transferencia de líneas de ensamble de Puerto Rico a Saltillo. Actualmente es profesor investigador del departamento de posgrado de COMIMSA y además forma parte del equipo de trabajo de la Celda de Manufactura Avanzada donde se encuentra trabajando en el desarrollo de algoritmos para la optimización de movimientos para el brazo robótico de ensamble. Su área de investigación se enfoca a la Inteligencia Artificial, la optimización, la creación de métodos híbridos y la creación de software.



L. Abril Torres-Méndez es profesora investigadora de tiempo completo en el Grupo de Robótica y Manufactura Avanzada del CINVESTAV Unidad Saltillo desde febrero del 2006. Sus áreas de interés son en la visión artificial y robótica móvil tanto terrestre como acuática. Su investigación se enfoca en el análisis estadístico de información visual para el mejoramiento y corrección del color, modelado de ambientes no estructurados, navegación autónoma, recono-

cimiento de objetos 3D, reconocimiento de gestos y seguimiento de la mirada en la interacción humano-robot.



Pedro Pérez-Villanueva

es Ingeniero Industrial y de Sistemas por la Universidad Autónoma de Coahuila y Maestro en Sistemas de Información por la misma Universidad. Obtuvo el Doctorado en Ingeniería Industrial y Sistemas de Manufactura dentro del Programa Interinstitucional de Ciencia y Tecnología (PICYT-CONACYT) sede COMIMSA. Cuenta con una certificación en Procesos de Manufactura y una en Manufactura Esbelta. Cuenta con 27 años de experiencia en el área de diseño y sistemas de manufactura participando en proyectos para Pemex, CFE, General Electric / Nuclear, STEWART & STEVENSON, así como en proyectos con plantas del sector automotriz. Sus líneas de investigación comprenden la aplicación de sistemas inteligentes y sistemas dinámicos en el diseño y manufactura de productos así como para el ensam-

ble y desensamble de estos. Actualmente es Profesor-Investigador en el PICYT sede COMIMSA y ocupa el cargo de subgerente Académico.