

Evolution-Based Vision Algorithm with Fuzzy Fitness Function for Obstacle Detection

Haythem Ghazouani, Tagina Moncef, René Zapata

► **To cite this version:**

Haythem Ghazouani, Tagina Moncef, René Zapata. Evolution-Based Vision Algorithm with Fuzzy Fitness Function for Obstacle Detection. META: Metaheuristics and Nature Inspired Computing, Oct 2010, Djerba, Tunisia. pp.51-52. lirmm-00954358

HAL Id: lirmm-00954358

<https://hal-lirmm.ccsd.cnrs.fr/lirmm-00954358>

Submitted on 1 Mar 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Evolution-Based Vision Algorithm with Fuzzy Fitness Function for Obstacle Detection

H. Ghazouani¹, M. Tagina² and R. Zapata¹

1. LIRMM, 161 rue Ada, 34392 Montpellier CEDEX 5, France
 haythemghz@yahoo.fr, zapata@lirmm.fr

2. LI3, Ecole Nationale des Sciences de l'Informatique, Tunis, Tunisie
 moncef.tagina@ensi.rnu.tn

Keywords: Parisian evolutionary approach, fuzzy correlation, obstacle detection, vision.

1 Introduction

The proposed algorithm is a fast evolution-based vision technique for real-time obstacle detection [1]. Based on the Parisian approach [2] [3], our algorithm evolves a population of 3D particles which constitutes a three-dimensional representation of the scene. Evolution is controlled by a fuzzy fitness function able to deal with uncertain camera measurements, and uses classical evolutionary operators. The result of the algorithm is a set of 3D particles gathered on the surfaces of obstacles.

2 Fuzzy correlation

Assuming that the images have been rectified to have disparities only along the Y-coordinate axis, a particle is defined as a 3D point (x, y, z) that projects to the pixel (x_p, y_p) in the reference image and the pixel (x_p, y_p+d) in the right image. If the particle is within an opaque object of the environment, then corresponding pixels in the two images will have similar contrasts and similar neighbouring as shown in Figure 1. Conversely, if the particle is within a heterogeneous surface, similarity between corresponding pixels neighbouring will be low.

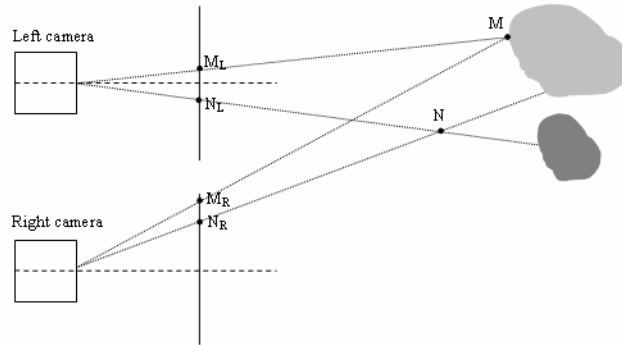


Figure 1. Pixels M_L and M_R are projections of particle M , are strongly correlated, pixels N_L and N_R receive light reflections from two objects with different contrasts.

We propose to calculate a fuzzy correlation cost to express the similarity between two corresponding pixels. We have defined three grey classes which are Black class, Average Class, and White class. Each one of the defined classes has its membership function. Let $\mu_{black}(m)$, $\mu_{average}(m)$ and $\mu_{white}(m)$ be the degrees of membership of the pixel m respectively to the black class, the average class and the white class. We have used Gaussian membership functions centred in 0, 255 and 127.5. Two pixels m_1 and m_2 are strongly correlated if they belong to the same class. This

proposition can be expressed using fuzzy logic as follow:
$$FC(m_1, m_2) = \max \left(\begin{array}{l} \min[\mu_{black}(m_1), \mu_{black}(m_2)], \\ \min[\mu_{average}(m_1), \mu_{average}(m_2)], \\ \min[\mu_{white}(m_1), \mu_{white}(m_2)] \end{array} \right).$$

Thereafter, we will use the notation: $FC(x, y, z) = FC(m_1, m_2)$ with $m_1 = (x_p, y_p)$ and $m_2 = (x_p, y_p + d)$ projection of the 3D point (x, y, z) . $FC(x, y, z)$ is the fuzzy correlation cost of the projections of the particle (x, y, z) into the reference and the right images.

3 Genetic conception of the algorithm

We have used a fuzzy fitness function that tolerates uncertainty of measurements. To favour evolution of particles towards obstacle surfaces, we propose the fitness function: $Fitness(x, y, z) = \left| \nabla(x_p, y_p) \right| \times \left| \nabla(x_p, y_p + d) \right| \sum_{(i,j) \in N} FC(x_p + i, y_p + d + j)$ that characterizes similarity between support areas of the particle projections to the left and right images. N is a neighbouring introduced to have a discriminating comparison between the projections. $\left| \nabla(x_p, y_p) \right| \times \left| \nabla(x_p, y_p + d) \right|$ are Sobel gradient norms on left and right projections of the particle [4]. Individual chromosomes are 3D particle coordinates (x, y, z) . Population is randomly initialized within the field of view of the reference camera from a distance d_{min} to the camera. Depths are allocated uniformly by distributing the inverse of z between zero and $1/d_{min}$. *Selection* is elitist and deterministic. Bidimensional *sharing* penalize particles that project into overcrowd area of the image by reducing their fitness functions by $K \times N$; K is the sharing coefficient and N is the number of particles that projects in a sharing radius R around the current particle. *Mutation* operator that allow extensive exploration of search space uses an approximation of a Gaussian noise that will be added to (x, y, z) parameters. *Z* mutation is to add Gaussian noise to the inverse of z so that the variance of z mutation will be coherent with the density of particles that decreases with distance: $z_{new} = \frac{z \times d_{min}}{d_{min} + z \times noise}$.

Thereby, we have used mutations variances σ_x , σ_y , and σ_z equal to R to be in the same order as the mean distance between two neighbouring particles. Mutation of x and y is calculated in a way to have independence between variance of the displacement of the image of the particle and its depth: $x_{new} = x + (z_{old} \times noise) \times (d_{focal})^{-1}$. The noise chosen for x and y is Gaussian and its variance is in the same order as the projection of the distance of a particle to its nearest neighbour $\sigma_x = \sigma_y = \sqrt{N_{pixels} / N_{pop}}$. We have defined a *barycentric crossover*. Given two parents M and N , the algorithm gives their offspring F as: $\overline{OF} = \lambda \overline{ON} + (1 - \lambda) \overline{OM}$ (λ is randomly chosen in $[0, 1]$). An example of execution in a robot environment scene is illustrated in Figure 2.

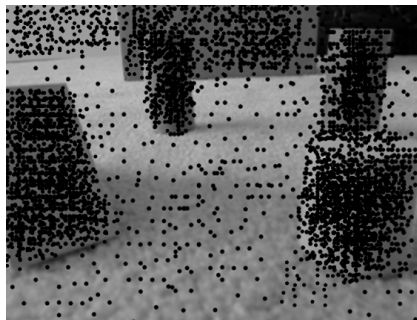


Figure 2. Convergence of our algorithm after 100 generations.

References

- [1] S. Cagnoni, (2008), Evolutionary Computer Vision: A Taxonomic Tutorial, HIS.
- [2] G. Olague and C. Puente, (2006). Parisian evolution with honeybees for three-dimensional reconstruction," Genetic And Evolutionary Computation Conference.
- [3] E. Dunn, G. Olague, and E. Lutton, (2006). Parisian camera placement for vision metrology, Pattern Recognition Letters, vol. 27, pp. 1209-1219.
- [4] R. C. Jain, R. Kasturi, and B. G. Schunck (1994). Machine Vision, McGraw-Hill: New York.