

## Research Article

# Pose Estimation in Noncontinuous Video Sequences Using Evolutionary Correlation Filtering

**Kenia Picos** <sup>1</sup>, **Ulises Orozco-Rosas** <sup>1</sup>, **Victor H. Díaz-Ramírez**,<sup>2</sup> and **Oscar Montiel** <sup>2</sup>

<sup>1</sup>*CETYS Universidad, Centro de Innovación y Diseño (CEID), Ave. CETYS Universidad No. 4, El Lago, 22210, Tijuana, BC, Mexico*

<sup>2</sup>*Instituto Politécnico Nacional, CITEDI-IPN, Ave. Instituto Politécnico Nacional 1310, Tijuana 22435, BC, Mexico*

Correspondence should be addressed to Kenia Picos; [kenia.picos@cetys.mx](mailto:kenia.picos@cetys.mx)

Received 27 May 2018; Revised 24 September 2018; Accepted 18 October 2018; Published 31 October 2018

Academic Editor: Ioannis Kostavelis

Copyright © 2018 Kenia Picos et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, we propose an evolutionary correlation filtering approach for solving pose estimation in noncontinuous video sequences. The proposed algorithm computes the linear correlation between the input scene containing a target in an unknown environment and a bank of matched filters constructed from multiple views of the target and estimates of statistical parameters of the scene. An evolutionary approach for finding the optimal filter that produces the highest matching score in the correlator is implemented. The parameters of the filter bank evolve through generations to refine the quality of pose estimation. The obtained results demonstrate the robustness of the proposed algorithm in challenging image conditions such as noise, cluttered background, abrupt pose changes, and motion blur. The performance of the proposed algorithm yields high accuracy in terms of objective metrics for pose estimation in noncontinuous video sequences.

## 1. Introduction

Pose estimation is an important task widely used in three-dimensional (3D) imaging applications to obtain descriptors such as location, orientation, scaling, depth, or geometric visualization of a target [1]. Applications, such as object tracking, fixture alignment, camera-based optical metrology, and augmented reality, are addressed by performing pose estimation research [2, 3]. The problem of 3D pose estimation presents high complexity when using monocular images, in which depth information of the target within a scene is unknown [4, 5]. The pose is characterized by how the object is viewed through an observer (camera) that, in turn, is determined by the estimation of its location, orientation, and scaling parameters [6]. An effective pose recognition system must be able to produce low approximation errors between the real and estimated parameters [7]. Furthermore, it is important to consider that real images are commonly degraded by challenging conditions such as noise, background clutter, motion blur, and geometrical changes of the target. Generally, these challenges compromise the system's performance by increasing the probability of occurrence of high estimation errors. Thus, it is required an effective

and robust approach to solve pose estimation under image degradations.

Correlation filtering is a pattern recognition technique that presents high accuracy in location estimation of a target [8]. This technique is given by a linear system whose frequency response is designed to produce a high matching value between a reference image of the target and the input scene [9]. Correlation filtering also provides high efficiency in target detection under noisy environments [10, 11].

Conventionally, the design of correlation filters requires an explicit knowledge of the appearance and shape of the target [12]. An effective design strategy consists of the construction of a set of correlation filters, in which each filter is modeled with each possible appearance version of the target that is expected in the observed scene [12]. Thus, this approach allows us to recognize several geometrically modified versions of the target.

Template matching based on correlation filters can be used to solve 3D pose estimation of rigid objects [13]. Moreover, the pose estimation problem can be modeled as a search problem, in which the goal is to find the reference target view that gives the best match between the actual view of the target in the scene [14]. By using a template matching

approach, a big set of correlation filters can be required to find a good match [15]. The design and construction of correlation filters is a high-dimensional complex problem, thus finding an optimal solution will require a very large search space. This strategy presented a narrow exploration space using a local search algorithm [13]. In this strategy it was assumed that the object would appear with smooth pose transitions through the video frames. However, in case of abrupt pose changes the algorithm would require performing an extended and time-consuming search. It should be noted that, in the case of an extended search, the number of iterations will increase to obtain a good estimate, i.e., the optimum. In consequence, the risk of getting stuck in a local optimum will be increased also.

In this work, we are encouraged finding the optimal solution for 3D pose estimation problem when the target can present abrupt pose changes in the scene, i.e., when the search space of location, orientation, and scaling parameters of the target is very big. This challenge needs to be treated as a combinatorial optimization problem that can become numerically intractable as the number of feasible poses of the target increases [16, 17]. From the point of view of computational complexity, this problem is classified as an NP-hard problem [18], since it will require an algorithm with the ability to solve problems in nondeterministic polynomial (NP) time, but the solution can be verified in polynomial time. Moreover, it is at least as difficult as the hardest problems in NP [19, 20].

In this paper, we propose an evolutionary correlation filtering approach to solve the pose estimation problem efficiently. The proposed evolutionary correlation filtering approach can be defined as a hybrid metaheuristic that combines correlation filters [21] and evolutionary computation [22]. Specifically, the proposed evolutionary correlation filtering approach employs a genetic algorithm [23] because of its computational efficiency, accuracy, robustness, and simplicity for implementation [24, 25]. We propose a 3D pose estimation algorithm for video sequences that contain a target with abrupt pose changes across frames while the input scenes are degraded by challenging conditions, such as the presence of additive noise, cluttering, and motion blur. The proposed algorithm utilizes an evolutionary approach to perform adaptive correlation filtering for 3D pose estimation, in which a bank of filters can evolve to obtain high estimation accuracy in 3D pose parameters of a target from monocular scenes. Based on the above considerations, the main motivation behind this work is to present an accurate algorithm for pose estimation that can be employed in critical applications of 3D imaging and object tracking. The main contributions of this work can be summarized as follows:

- (i) The three-dimensional pose parameters of a target pose can be estimated with a single monocular camera.
- (ii) The proposed algorithm can solve the pose estimation problem with evolutionary correlation filtering using a hybrid approach between a genetic algorithm and correlation filters.

- (iii) An adaptive filter bank dynamically evolves for pose tracking of a target by presenting high estimation accuracy in terms of location, orientation, and scaling.
- (iv) Robust pose estimation is performed under noncontinuous video sequences, with image conditions such as additive noise, cluttered background, and motion blur.

The paper is organized as follows. Section 2 presents the theoretical framework of correlation filtering for object recognition and evolutionary computation for global optimization. Section 3 describes our proposed method for pose estimation using evolutionary correlation filtering. In Section 4, we present and discuss the experimental results by performance evaluations in noncontinuous video sequences. Finally, the conclusions are summarized in Section 5.

## 2. Theoretical Framework

In this section, a brief description of the main components of the proposed evolutionary correlation filtering approach is presented. This approach combines correlation filters and a metaheuristic based on a genetic algorithm as the global optimization method to solve the 3D pose estimation problem.

*2.1. Object Recognition with Correlation Filters.* Assume that a 3D object is observed with a monocular camera, as shown in Figure 1. The captured frame  $f(\mathbf{x})$  consists on a projection of the 3D scene into the plane  $\pi \in \mathbb{R}^2$ . The scene is composed of an object  $t(\mathbf{x})$  which is placed in an unknown location  $\mathbf{d} = [\delta_x, \delta_y]^T$  and embedded into a disjoint background  $b(\mathbf{x})$ , as follows:

$$f(\mathbf{x}) = t(\mathbf{x} - \mathbf{d}; \mathbf{\Gamma}) w(\mathbf{x} - \mathbf{d}; \mathbf{\Gamma}) + b(\mathbf{x}) \bar{w}(\mathbf{x} - \mathbf{d}; \mathbf{\Gamma}) + \eta(\mathbf{x}), \quad (1)$$

where  $\mathbf{x} = [X\rho/\rho_0, Y\rho/\rho_0]^T$  represent Cartesian coordinates in the image plane, which are the mapped coordinates from  $\mathbb{R}^3 \mapsto \mathbb{R}^2$ , and  $[X, Y, Z]^T$  represent Cartesian coordinates in the 3D space. As shown in Figure 1, the term  $\rho$  is the distance from  $\pi$  to the reference coordinate system of the 3D scene, and  $\rho_0 = \rho + Z$  is the total distance from the center of projection (COP) of the camera to the observed world. The term  $w(\mathbf{x})$  is a binary function that represents the support region of the target, given by

$$w(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{x} \in t(\mathbf{x} - \mathbf{d}; \mathbf{\Gamma}) \\ 0, & \text{elsewhere.} \end{cases} \quad (2)$$

Furthermore,  $\bar{w}(\mathbf{x}) = 1 - w(\mathbf{x})$  represents the inverse support region of the target. The additive noise denoted by  $\eta(\mathbf{x})$  is given by a zero-mean Gaussian distribution process. Moreover,  $\mathbf{\Gamma}$  in (1) is a transformation matrix that involves the appearance modifications of the target [13] related to scaling  $\mathbf{S}$  and rotation  $\mathbf{R}$  (with  $\phi, \theta, \varphi$  orientation parameters). Hence,  $\mathbf{\Gamma} = \mathbf{S} \mathbf{R}$  is related to the space  $\Pi \in \mathbb{R}^3$ .

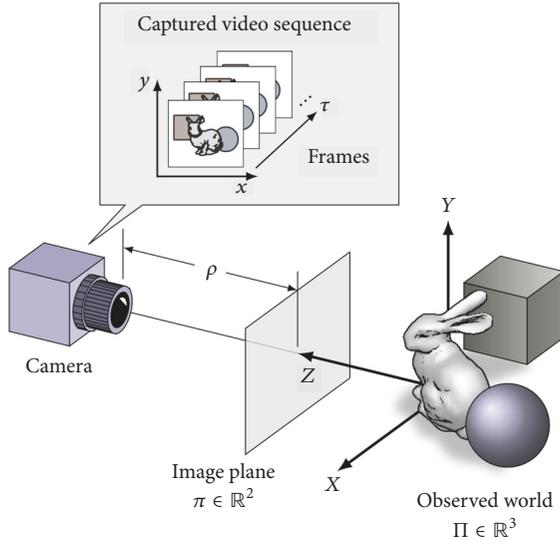


FIGURE 1: Image model representation.

Correlation filtering quantifies the level of linear correspondence between two signals. This filtering is widely used in pattern recognition applications to detect a target  $t(\mathbf{x})$  and estimate their location coordinates from an input scene  $f(\mathbf{x})$ . The correlation filtering can be carried out in frequency domain as follows:

$$c(\mathbf{x}) = f(\mathbf{x}) \otimes h(\mathbf{x}) = IFT \{F(\mathbf{u})H^*(\mathbf{u})\} \quad (3)$$

where  $F(\mathbf{u})$  and  $H(\mathbf{u})$  are the Fourier transforms of the input scene  $f(\mathbf{x})$  and the impulse response of a designed correlation filter  $h(\mathbf{x})$ , respectively, and  $\mathbf{u} = [\mu, \nu]^T$  are coordinates in the frequency domain. The impulse response of the correlator in (3) must be appropriately designed to produce a high correlation value in the area occupied by the target in the scene and a value close to zero elsewhere. Correlation filters can be modeled mathematically by the optimization of different performance criteria. For real-life applications, the signal-to-noise-ratio (SNR) is an appropriate design criterion for target recognition in noisy scenes. The generalized matched filter (GMF) is an optimal filter with respect to the SNR [26]. This filter is robust in the presence of additive noise and clutter. Additionally, it presents high accuracy in estimating the location of a target in the scene [27, 28]. The frequency response of the GMF is given by

$$H^*(\mathbf{u}) = \frac{T(\mathbf{u}) + m_b \overline{W}(\mathbf{u}) + m_t W(\mathbf{u})}{(1/2\pi) |\overline{W}(\mathbf{u})|^2 * N_b^0(\mathbf{u}) + (1/2\pi) |W(\mathbf{u})|^2 * N_\eta^0(\mathbf{u})}, \quad (4)$$

where  $T(\mathbf{u})$ ,  $W(\mathbf{u})$ , and  $\overline{W}(\mathbf{u})$  represent the Fourier transforms of the signals  $t(\mathbf{x})$ ,  $w(\mathbf{x})$ , and  $\overline{w}(\mathbf{x})$ , respectively. Moreover,  $m_t$  and  $m_b$  are the mean value of the target  $t(\mathbf{x})$  and background  $b(\mathbf{x})$ , respectively. The terms  $N_b^0$  and  $N_\eta^0$  denote spectral density functions of the zero-mean background

$b_0(\mathbf{x}) = b(\mathbf{x}) - m_b$  and the additive noise process  $\eta(\mathbf{x})$ , respectively [28].

Note that the filter given in (4) can be matched to a unique view of the target  $T(\mathbf{u})$  having specific pose parameters. So, to recognize a target which appears in many pose configurations, several views of the target can be used as reference templates for the filter design. In correlation filtering, often a bank of filters is used to store a set of references as the training set. Since the pose of the target in the scene is unknown, the bank of filters needs to contain one reference view of the target per each pose configuration that is expected in the scene. However, in real applications, having identical references in the training set is very complicated and, in many cases, unfeasible. So, to avoid this limitation we propose an evolutionary correlation filtering, in which the filter bank is adapted dynamically using evolutionary optimization. In this manner, a good pose estimate can be obtained by considering only a subset of all possible filters. Note that, within this proposal, the design of the filter bank is crucial to achieve a high matching score between the input scene and the reference for estimating the best pose state.

In the filter bank, a set of synthetic references are generated from a digital 3D model using computer graphics to establish a specific pose state. Each reference template represents a single view of the target  $t(\mathbf{x})$  having a unique pose configuration. Once the set of template references  $t_j(\mathbf{x})$  has been generated, a bank of filters  $\{H_j(\mathbf{u}); j = 1, \dots, N\}$  is computed with the help of (4). In Figure 2, the correlation operation is performed between the input scene  $f(\mathbf{x})$  and each filter in the bank. A correlation plane  $c_j(\mathbf{x})$  is obtained for the  $j$ -th filter of the bank. For each correlation plane, different matching scores with several references can be obtained for a given input scene. The highest correlation value represents the best match. The location of the target is calculated from the correlation plane given by (3). The coordinates of the target in the scene can be estimated as the coordinates of the highest peak in the correlation plane, as follows:

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmax}} \{|c(\mathbf{x})|^2\}, \quad (5)$$

where  $\hat{\mathbf{x}} = [\hat{x}, \hat{y}]^T$  represents the estimated location coordinates of the detected target within the input scene.

**2.2. Evolutionary Computation for Global Optimization.** Evolutionary computation is a family of population-based algorithms with a metaheuristic or stochastic optimization character for global optimization. It is inspired by biological evolution, where genetic algorithms are the most prominent example [29]. The genetic algorithms are based on the mechanisms of natural selection and natural genetics [30]. The robustness of these algorithms on complex problems has led to an increasing number of applications in the field of artificial intelligence, numerical and combinatorial optimization, computer science, and engineering [31].

The genetic algorithms became popular through the work of John Holland and particularly his book *Adaptation in Natural and Artificial Systems* [32]. Some of the advantages of genetic algorithms over other metaheuristics are flexibility

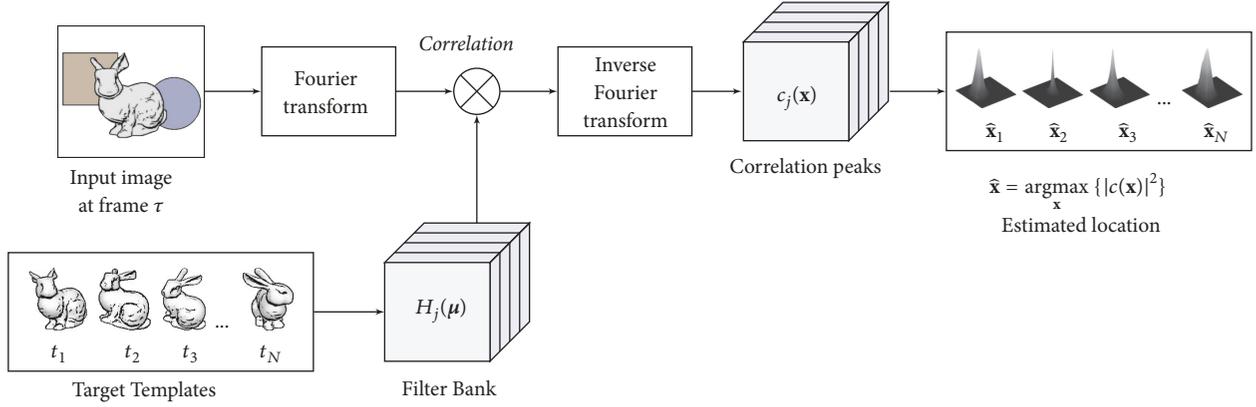


FIGURE 2: Block diagram of template matching using correlation filters.

in defining constraints and quality measures, the capability of working with both continuous and discrete variable, effectiveness in large search spaces, the power of providing multiple optimal/good solutions, and great potential for applying parallel computing techniques to speed up the computation [31].

Because of the above-mentioned advantages, we choose the genetic algorithm as the search tool for the pose estimation problem. In this work, the genetic algorithm is used to search the optimal parameters of the target's pose, consisting of scaling and orientation values.

A genetic algorithm is an adaptive heuristic search algorithm designed to simulate the evolution processes existing in natural systems, where, in the most basic form, a genetic algorithm can be modeled for computer simulation using the difference equation

$$P_{i+1} = \zeta(v(P_i)), \quad (6)$$

where  $i$  represents the generation counter (generations over time) and  $P_{i+1}$  is the new population obtained from the current population  $P_i$  after being modified by random variation  $v$  and selection  $\zeta$  [33].

Canonical genetic algorithms use a binary representation of chromosomes as fixed-length strings over the alphabet  $\{0, 1\}$ , such that they are well suited to handle pseudo-Boolean optimization problems of the form

$$\mathcal{F} : \{0, 1\}^\ell \longrightarrow \mathbf{R}. \quad (7)$$

In case of continuous parameter optimization problems, genetic algorithms typically represent a real-valued vector  $\mathbf{x} \in \mathbb{R}^n$  by a binary string  $\mathbf{y} \in \{0, 1\}^\ell$ , where the binary string is logically divided into  $n$  segments of equal length  $\ell'$  (i.e.,  $\ell = n \cdot \ell'$ ). Each segment is decoded to yield the corresponding integer value, and this integer value is in turn linearly mapped to the interval of real values [34]. In contrast to the binary codification, genetic algorithms with floating-point representation use a floating-point value as a parameter of the chromosome without performing coding and decoding processes [35]. For real-valued numerical optimization problems, floating-point representation outperforms binary

representations because they are more consistent, are more precise, and lead to faster execution [36].

### 3. Evolutionary Correlation Filtering Algorithm for Pose Estimation

In this section, we present the proposed algorithm for pose estimation of a target presented in noncontinuous video sequences. The main feature of this proposal is the robustness of the evolutionary correlation filtering when processing noisy and cluttered scenes. One of the most difficult challenges in target tracking is to track an object presenting appearance changes over time. The proposed algorithm is able to estimate and track accurately the pose of the target when presenting abrupt pose changes within a video sequence. The proposed evolutionary correlation filtering algorithm is modeled to achieve a robust pose estimation when facing noncontinuous video sequences. The proposed algorithm is based on correlation filters using an evolutionary computation approach. A genetic algorithm is employed to search for the best pose parameters in the computation of the filter bank in order to produce a high matching score. The genetic algorithm finds candidate solutions through the evolution of the pose parameters of the target. The bank of correlation filters is able to estimate the pose of the target under difficult image conditions, such as the presence of noise, motion blur, and cluttered background. Figure 3 shows the block diagram of the proposed algorithm for pose estimation based on correlation filtering and evolutionary computation. The proposed methodology is organized into three subsystems: observation, template matching, and filter bank evolution, which are explained below.

**3.1. Observation.** In the observation process, a video sequence enters into the system assuming the optical setup shown in Figure 1. An input scene (at frame  $\tau$ ) contains a target with an unknown pose, which is embedded in a cluttered background. In the case of real-life scenes, the image may be corrupted with additive noise introduced by the camera sensor. The signal model of the input scene  $f^{(\tau)}(\mathbf{x})$  is given by (1). The pose of the target is defined by

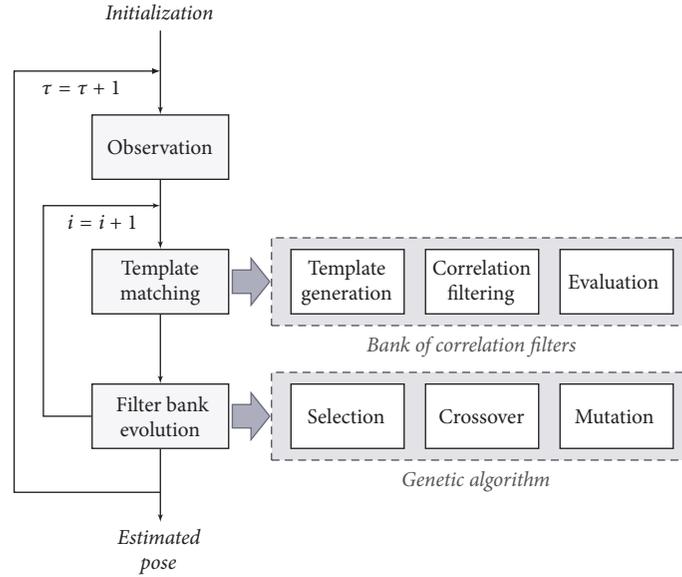


FIGURE 3: Proposed algorithm for pose estimation using evolutionary correlation filtering.

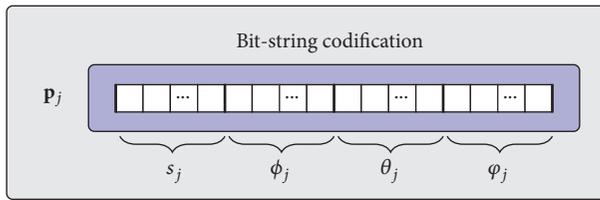


FIGURE 4: Bit-string representation of an individual.

the location parameters denoted by  $\mathbf{x} = [x, y]$  and by scaling and orientation parameters denoted by  $\mathbf{p} = [s, \phi, \theta, \varphi]$ . The state vector that contains the pose parameters of the target is given by

$$\mathbf{q} = [\mathbf{x}, \mathbf{p}] = [x, y, s, \phi, \theta, \varphi], \quad (8)$$

where  $\mathbf{q}$  represents the pose configuration for the current view of the target in the input scene, as shown in Figure 1. At the beginning ( $\tau = 0$ ), the algorithm performs an *initialization* process. This initialization consists in defining a set of different scaling and orientation configurations, represented by

$$P_i = [\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_{N_{pop}}]^T, \quad (9)$$

where the vector  $\mathbf{p}$  is given by a scaling and orientation configurations with parameters generated randomly using a uniform distribution. To solve the pose estimation problem, we apply the evolutionary correlation filtering. The individual (solution) representation consists of four parts as shown in Figure 4. The first part is a bit-string codification of the scaling and the second, third, and fourth parts are the bit-string codification of the angles  $\phi$ ,  $\theta$ , and  $\varphi$ , respectively.

For the consecutive frames  $\tau = \tau + 1$ , the population of individuals is evolved using the genetic operators. Therefore,

in each new frame, the initial population will be the population evolved in the last frame, so the initialization process is no longer required. Instead, the proposed algorithm guides the system into the evolved values of each pose parameters to converge in an accurate estimation. In the observation process, the input video sequence may be noncontinuous, which means that the pose of the target can be abruptly changed from one frame to another, caused by imprecise video capturing.

**3.2. Template Matching.** Template matching is employed to detect the target in the scene and estimate their location coordinates, within the current frame. As the main feature, the algorithm yields high accuracy in challenging image conditions such as noise, motion blur, and background clutter. In the proposed methodology, we implemented the template matching technique considering three different stages: template generation, correlation filtering, and evaluation.

In the *template generation* stage, a set of templates are used to construct a filter bank, in which each template is associated with an individual  $\mathbf{p}_j$ ,  $j = 1, 2, \dots, N_{pop}$ , where  $N_{pop}$  is the size of the population  $P_i$ . Each individual encodes a pose  $\mathbf{p}_j = [s_j, \phi_j, \theta_j, \varphi_j]$  given by the parameters of scaling and orientation of the target. The target templates  $t_j(\mathbf{x})$  are rendered using computer graphics. The  $j$ -th pose parameters of each template represent a candidate solution for the input frame. In the proposed methodology depicted in Figure 3, the initialization is given by the individuals generated randomly using a uniform distribution for the parameters of scaling and orientation. The estimated location parameters  $\hat{\mathbf{x}} = [\hat{x}, \hat{y}]$  are obtained from the output correlation plane. These parameters are given by the coordinates of the highest correlation peak, as shown in (5).

Once the templates are generated, the next stage is the computation of *correlation filtering*. For each generated template a matched filter  $H(\mathbf{u})$  is designed using (4) and

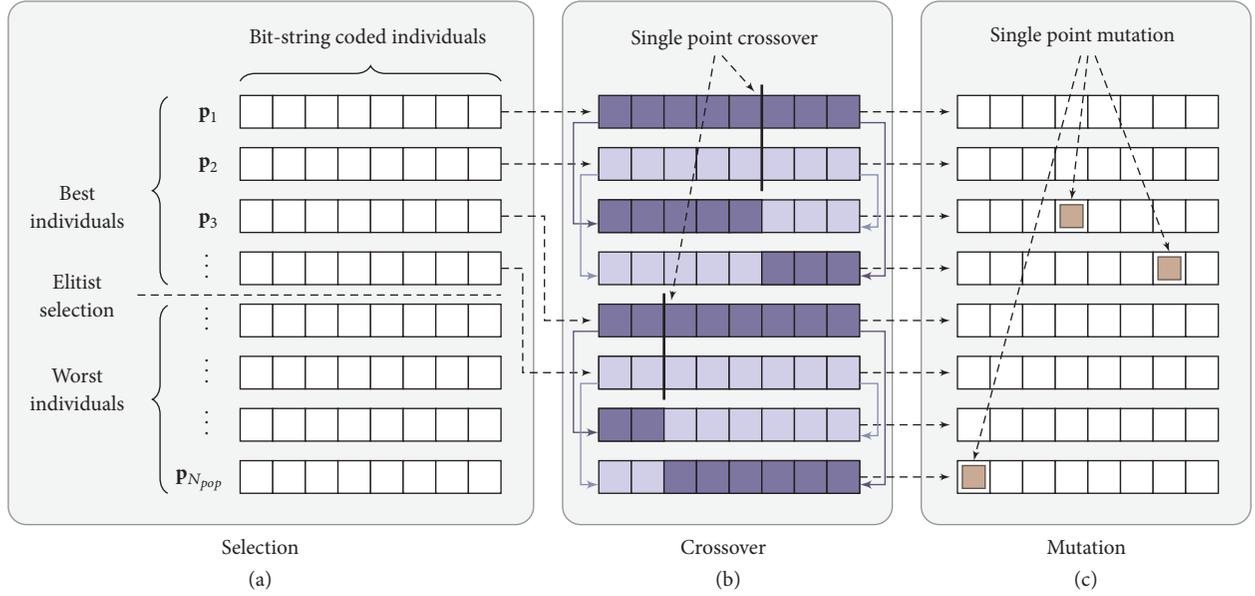


FIGURE 5: Genetic operators performed by the filter bank evolution subsystem.

by estimating the statistical parameters of the scene. The cross-correlation between the input frame  $f(\mathbf{x})$  and the impulse response of a filter is used for target detection. This operation is computed for each filter in the bank  $\{H_j(\mathbf{u}); j = 1, \dots, N_{pop}\}$ , as shown in Figure 2. Let  $c_j(\mathbf{x})$  be the correlation plane obtained by processing the observed image with the  $j$ -th filter of the bank. Then, the *evaluation* stage quantifies the ability of a filter to recognize between a target and a false object. This is done by utilizing the discrimination capability (DC) metric [27] given by

$$DC = 1 - \frac{|c_b(\mathbf{x})|^2}{|c_t(\mathbf{x})|^2}, \quad (10)$$

where  $c_t(\mathbf{x})$  represents the maximum correlation value in the area of the target in the input frame and  $c_b(\mathbf{x})$  is the maximum correlation sidelobe generated outside the area occupied by the target. A DC value close to unity indicates a reliable detection of the target. Instead, negative DC values indicate that the filter is unable to recognize the target. For target detection, a specific threshold  $\delta_d$  can be established, in which values of  $DC > \delta_d$  can be determined as a detection, and values of  $DC \leq \delta_d$  are without detection.

In the next subsystem, the DC is utilized as an objective function to quantify the quality (fitness value) of the candidate solutions contained in  $P_i$ . The fundamental approach for optimization is to formulate a single standard of measurement, a cost function, that summarizes the performance or fitness value of a decision and iteratively improve this performance through available alternatives [36]. In this work, the pose estimation problem is defined and captured in the objective function depicted by (10) that indicates the quality of the individuals, i.e., the fitness of any potential solution.

**3.3. Filter Bank Evolution.** Figure 5 shows a general view of the genetic operators employed by the filter bank evolution subsystem to find the best scaling and orientation parameters of the target. Once the initial population  $P_i$  has been created in the observation subsystem and it has been evaluated to obtain the fitness value of each individual in the template matching subsystem, we continue with the filter bank evolution, where the selection stage is the first step.

The *selection* stage drives the genetic algorithm to improve the population fitness over the successive generations [37]. In the selection stage, the best individuals are chosen according to their fitness value. The higher the fitness value is, the more the chance an individual has to be selected. Hence, selection tends to eliminate those individuals that present a weak fitness value. The selection mechanism determines which individuals will be retained as “parents” for being used in the crossover stage, and the genetics underlying their behavioral traits are passed to their offspring [33]. In this work, the strategy of elitist selection is applied over the population, where the individuals with the best fitness values (best individuals of the population) are chosen as parents guaranteeing a place in the next stage, as illustrated in Figure 5(a). An asymptotic convergence is guaranteed by employing the elitist selection, which always retains the best individuals in the population [36].

Next, the *crossover* stage is performed over the population. Holland [32] indicates that crossover provides the primary search operator. The crossover operator randomly chooses a locus and exchanges the subsequences before and after that locus between two individuals to create two offspring. For example, the strings 11111111 and 00000000 could be crossed over the fifth locus in each to produce the two offspring 11111000 and 00000111. The crossover operator roughly mimics biological recombination between two

single-chromosomes organisms [29]. In this work, single-point crossover is employed. The selected individuals generate “offspring” via the use of crossover operator. Crossover is applied to two individuals called “parents”, which create two new individuals called “offspring”. This is done by selecting a random position along the coding and splicing the section that appears before the selected position. A new individual will be formed by the first string with the section that appears before the selected position of the first parent and the second string with the section that appears after the selected position of the second parent and vice versa for the second new individual (see Figure 5(b)).

Then, the *mutation* stage is performed over the population. The mutation operator randomly alters a certain percentage of the bits in the list of individuals. Mutation is the second way in which a genetic algorithm explores a cost surface. This can introduce traits which are not in the original population; therefore, it keeps the genetic algorithm from converging prematurely before sampling the entire cost surface. A single-point mutation changes a value of 1 to 0 and vice versa. In this work, mutation points are randomly selected from the total population size, as illustrated in Figure 5(c). Increasing the number of mutations, the algorithm freedom to search outside the current region of the variable space also increases. It also tends to distract the algorithm from converging on local minima [37].

The filter bank evolution subsystem iterates until the maximum number of generations  $N_{gen}$  is reached. Therefore, the genetic algorithm evolves the scaling and orientation parameters that are codified in each individual  $\mathbf{p}_j$  to obtain the corresponding optimal values for the pose. Finally, the estimated pose  $\hat{\mathbf{q}}$  is determined by the best candidate solution from the last iteration of the genetic algorithm, given by

$$\begin{aligned} \hat{\mathbf{q}} &= \left\{ (\mathbf{q} \in \Gamma) \sim H_{j^*}(\boldsymbol{\mu}) : j^* \right. \\ &= \left. \arg \max_j \{DC_j; j = 1, \dots, N_{pop}\} \right\}. \end{aligned} \quad (11)$$

It should be noted that the estimated pose  $\hat{\mathbf{q}}$  is used to construct the filter that will recognize the target in the scene with the highest DC value, which is, in turn, the estimated pose state of the current frame.

#### 4. Experimental Results

In this section, we present the experimental results obtained with the proposed algorithm for pose estimation in non-continuous video sequences. We evaluate the convergence of the genetic algorithm in terms of the detection efficiency of the evolutionary correlation filters. Also, we quantify the performance of pose estimation in terms of location and orientation errors. The performance of the proposed algorithm is evaluated and discussed by processing noisy input scenes. Then, candidate solutions given by the genetic algorithm are analyzed in terms of quality of pose estimation. We measure the performance of the proposed approach

TABLE 1: Parameters used in the experiments.

Parameters	Values
Population size	16, 32, 64, 128
Number of generations	1, 10, 20, 50, 100, 200, 500
Selection rate	0.5
Mutation rate	0.2

under abrupt pose changes of the target. Finally, we show that the proposed algorithm performs well in noncontinuous video sequences in a real environment.

The proposed evolutionary correlation filtering algorithm for pose estimation was implemented in C/C++ programming language. Computer graphics are computed with OpenGL library. We used the Stanford Bunny [38] as a reference model which was rendered with the Phong’s lighting model [39] with soft reflections for generation of the synthetic templates. The utilized 3D model contains 35,947 vertices and 69,451 triangles. For real video evaluations, we use a 3D printer for the construction of a real object with dimensions of 4.5×5.0×3.0 cm. The tested video sequence contains 400 frames of 256×256 pixels and RGB color channels, captured by a monocular camera.

Table 1 shows the parameters used in the evolutionary correlation filtering algorithm for the experiments. Each test was executed 30 times for repeatability with different random seeds used to create the initial population. The number of generations was varied to see the effect in the convergence. For this, we employed from 1 to 500 generations for testing. Additionally, we varied the number of individuals in the population using from 16 to 128 individuals. The mutation parameter was set to 0.2, which means that the random mutations alter the 20% of the bits in the list of individuals. The elitist selection parameter was set to 0.5; i.e., the elitist selection is performed where the 50% of the individuals with the best fitness values are chosen as parents guaranteeing a place in the next generation.

*4.1. Performance Results of the Filter Bank Evolution.* We tested the performance of the algorithm in terms of discrimination capability (DC) yielded by the correlation filtering through generations of the evolutionary algorithm, as shown in Figure 6. With 95% of confidence, we can see that the algorithm exhibits a high detection efficiency by producing a DC= 0.9802 ± 0.002 when using 500 generations and a population of 128 individuals. This result indicates that the proposed algorithm presents a reliable performance in terms of target detection.

Also, in this experiment we tested the performance of the proposed algorithm in terms of accuracy of the location estimation of the target. For this we measure the location error (LE) between the real  $\mathbf{x}$  and estimated  $\hat{\mathbf{x}}$  coordinates as follows:

$$LE = \|\mathbf{x} - \hat{\mathbf{x}}\|. \quad (12)$$

We varied the number of individuals in the population per generations of the evolutionary correlation filtering algorithm to measure the evolution behavior. Figure 7 presents

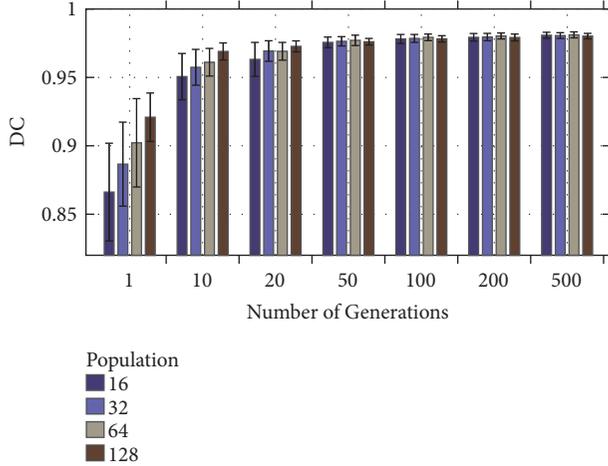


FIGURE 6: Performance of target detection of the evolutionary correlation filtering algorithm in terms of DC. The solid and error bars represent the mean value and the standard deviation, respectively, for 400 frames in 30 separate sample runs.

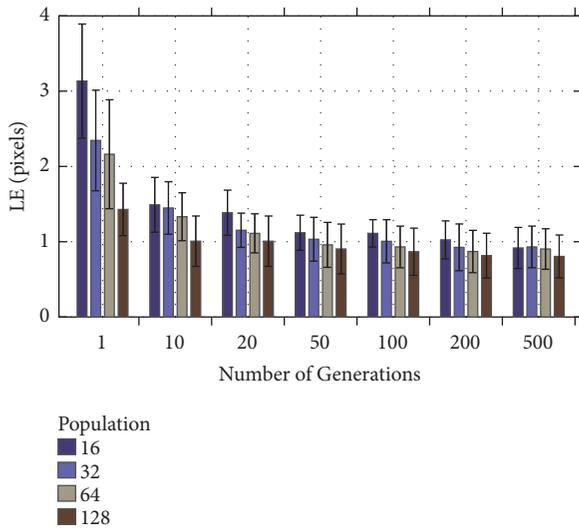


FIGURE 7: Performance of location estimation of the proposed evolutionary correlation filtering algorithm in terms of LE. The solid and error bars represent the mean and the standard deviation, respectively, for 400 frames in 30 separate sample runs.

the results in terms of LE. With 95% of confidence, the algorithm yields  $LE = 0.80 \pm 0.28$  pixels using the maximum number of generations with 128 individuals. The orientation error (OE) between the real and estimated pose can be computed as

$$OE = \|\Phi - \hat{\Phi}\|, \quad (13)$$

where  $\Phi = (\phi, \theta, \varphi)$  are the true orientation angles of the target and  $\hat{\Phi} = (\hat{\phi}, \hat{\theta}, \hat{\varphi})$  are the estimated angles for each orientation component. As shown in Figure 8 the accuracy of orientation estimation of the target is measured in terms of the OE. When using 500 generations and the

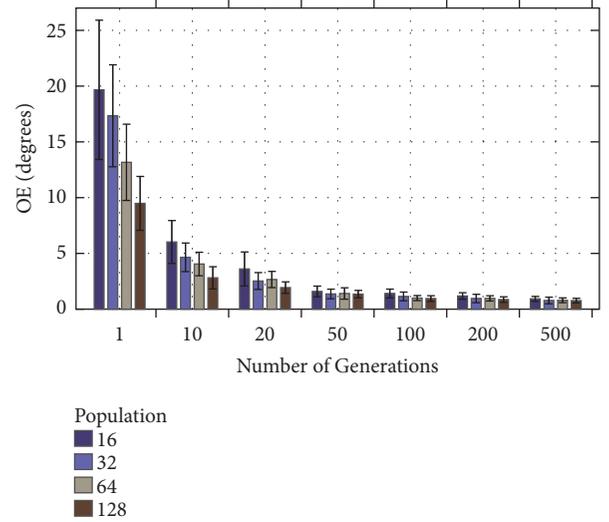


FIGURE 8: Performance of pose estimation of the proposed evolutionary correlation filtering algorithm in terms of OE. The solid and error bars represent the mean and the standard deviation, respectively, for 400 frames in 30 separate sample runs.

maximum population size (128 individuals), the evolutionary correlation filtering algorithm reduces the orientation error. The proposed algorithm yields high accuracy in pose estimation by producing  $OE = 0.76 \pm 0.21$  degrees with 95% of confidence. This result demonstrates that our proposed method is accurate and robust for pose estimation given in terms of location and orientation errors.

**4.2. Performance Results in Noisy Conditions.** The performance of the proposed algorithm is evaluated when processing scenes degraded with additive noise. We tested the noise robustness of the algorithm by considering additive noise with signal-to-noise-ratio (SNR) values of 50 dB, 10 dB, 5 dB, and 2 dB with respect to the number of generations of the evolutionary correlation filtering algorithm. For this experiment, we use a fixed population size of 32 individuals and 20% of mutation rate.

Figure 9 presents the results for noisy scenes given in terms of the DC. With 95% of confidence, the detection efficiency of the proposed algorithm is  $DC = 0.92 \pm 0.09$  for 50 dB and 10 dB SNR. Moreover, for highly noisy conditions of 5 dB and 2 dB SNR, the detection efficiency of the algorithm is  $DC = 0.87 \pm 0.13$ . We can see the reliability of the correlation filtering increments in terms of DC through generations of the evolutionary algorithm.

Figure 10 shows the performance in terms of LE for different additive noise SNR levels. With 95% of confidence, the accuracy of the location estimation of the target produced by the proposed algorithm is  $LE = 2.03 \pm 2.21$  pixels in the case of the additive noise SNR values of 5 dB and 2 dB. For additive noise SNR values of 50 dB and 10 dB, the proposed algorithm yields  $LE = 1.33 \pm 0.68$  pixels.

The performance of the proposed algorithm in terms of orientation estimation of the target is shown in Figure 11.

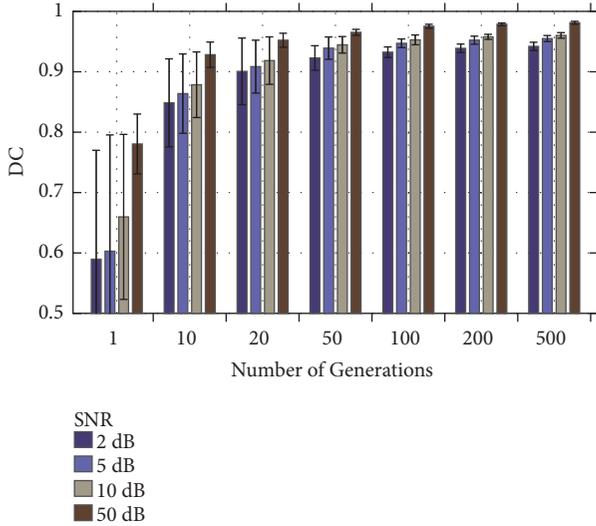


FIGURE 9: Performance of target detection of the proposed algorithm for noisy scenes in terms of the DC. The solid and error bars represent the mean value and the standard deviation, respectively, for 400 frames in 30 separate sample runs.

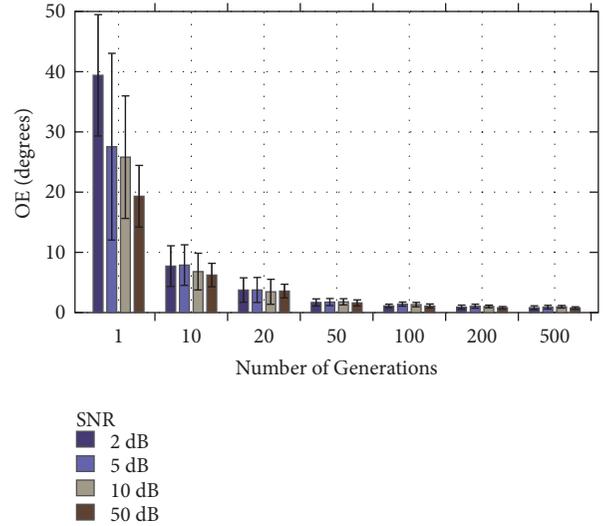


FIGURE 11: Performance of pose estimation of the proposed algorithm for noisy scenes in terms of the OE. The solid and error bars represent the mean value and the standard deviation, respectively, for 400 frames in 30 separate sample runs.

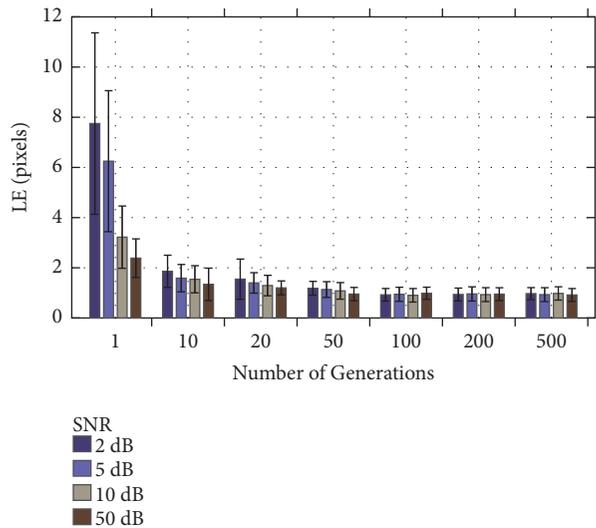


FIGURE 10: Performance of location estimation of the proposed algorithm for noisy scenes in terms of the LE. The solid and error bars represent the mean value and the standard deviation, respectively, for 400 frames in 30 separate sample runs.

For highly noisy conditions of 5 dB and 2 dB SNR, the proposed algorithm yields  $OE = 7.12 \pm 11.89$  degrees. For noisy conditions 10 dB of SNR, the performance is  $OE = 5.32 \pm 7.87$  degrees with 95% of confidence. Accordingly, with the obtained results, we can see that despite the presence of additive noise in the scene the performance of the algorithm gradually grows as the number of generations of the evolutionary correlation filtering algorithm increases. It should be noted that the proposed algorithm is able to produce low estimation errors of  $OE = 1.19 \pm 0.35$  degrees after fifty generation of the evolutionary algorithm.

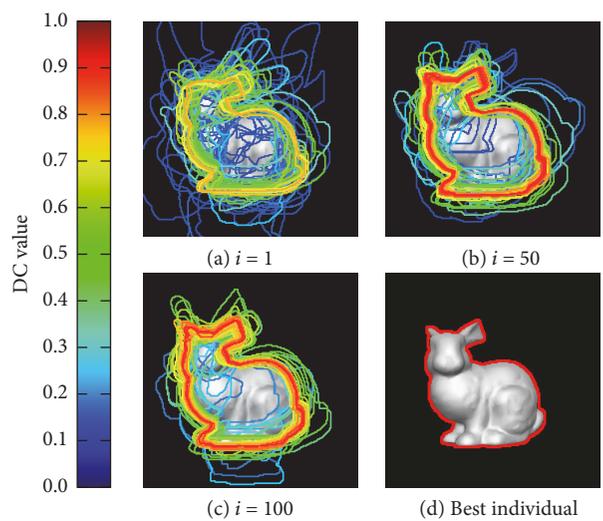


FIGURE 12: Candidate solutions for 1, 50, and 100 generations, and the best solution obtained by the proposed evolutionary correlation filtering approach.

**4.3. Evolution of Pose Estimation Results through Generations.** Figure 12 shows examples of the candidate solutions obtained with the proposed algorithm. For each solution, the DC is computed to evaluate the quality of the template matching. As we can see in Figure 12(a) candidate templates are presented for the first generation of the evolutionary correlation filtering algorithm. Then, the algorithm evolves to a better pose estimation given in terms of DC value. Figures 12(b) and 12(c) present the evolution of the algorithm after 50 and 100 generations achieving high target recognition performance of  $DC = 0.958 \pm 0.039$ . Figure 12(d) shows the best solution given by the proposed algorithm for the current frame.

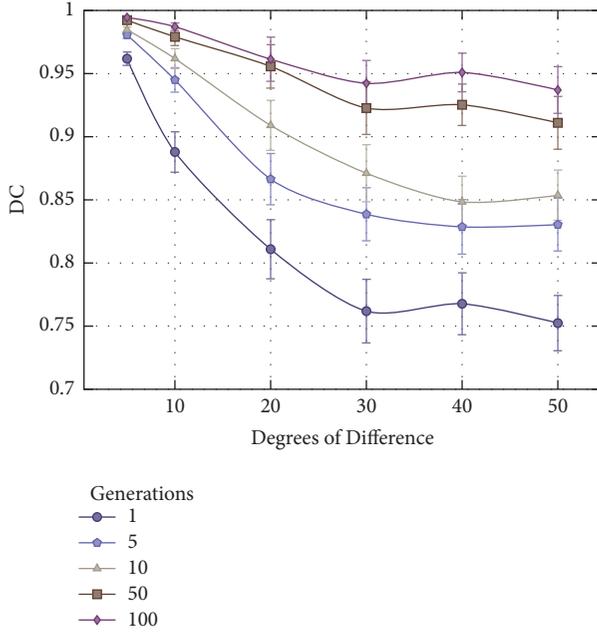


FIGURE 13: Results of target detection of the proposed algorithm in noncontinuous video sequences in terms of DC. The error bars represent the standard deviation of measurements for 400 frames in 30 separate sample runs.

**4.4. Performance Evaluation in Noncontinuous Video Sequences.** We test the robustness of the proposed algorithm in frame sequences which contain noncontinuous pose changes of a target. For these experiments, we process a synthetic video composed of abrupt pose changes of a target varying from 5 to 50 degrees between two consecutive frames. The pose changes are calculated in terms of the degrees of difference given by the Euclidean distance for the three coordinate angles of the current ( $\tau$ ) and the previous frame ( $\tau - 1$ ), given by  $|\theta^{(\tau)} - \theta^{(\tau-1)}|$ , where  $\theta$  denotes the orientation of the target in the three coordinate angles.

Figure 13 shows the performance of the proposed algorithm in terms of the DC for input frame sequences containing abrupt pose changes. With 95% of confidence, the algorithm produces a DC =  $0.95 \pm 0.04$  for sequences equal to or less than 20 degrees of difference between consecutive frames. From 30 to 50 degrees of pose variation from one frame to another, the performance of pose estimation of the algorithm slightly decreases to DC =  $0.86 \pm 0.07$ .

Figures 14 and 15 show the performance of the proposed algorithm in terms of location and orientation errors, respectively. With 95% of confidence, the proposed algorithm yields LE =  $1.63 \pm 0.95$  pixels when pose changes of 20 degrees among consecutive frames are allowed. Starting from 30 degrees of difference, the location error increases to LE =  $4.10 \pm 2.00$  pixels. In terms of orientation estimation, the algorithm obtains a pose estimation accuracy of OE =  $9.21 \pm 4.55$  degrees in sequences where pose changes of 20 degrees or less are allowed. For the case of 50 degrees of difference, the orientation error increases as expected; however, it can be

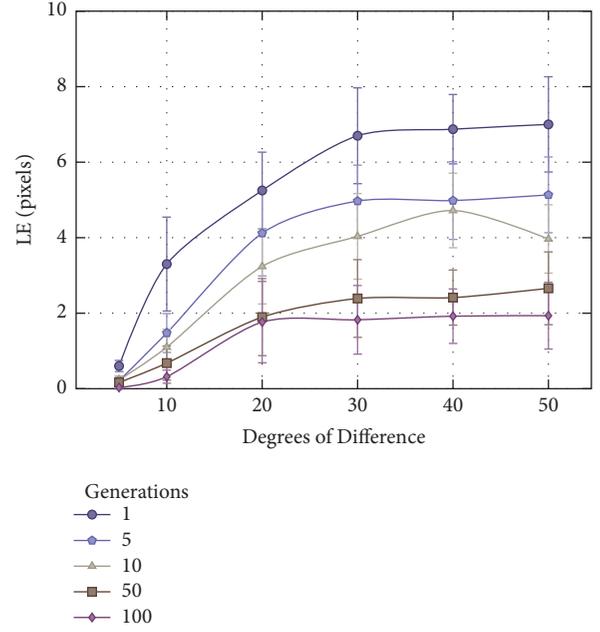


FIGURE 14: Results of location estimation of the proposed algorithm in noncontinuous video sequences in terms of LE. The error bars represent the standard deviation of measurements for 400 frames in 30 separate sample runs.

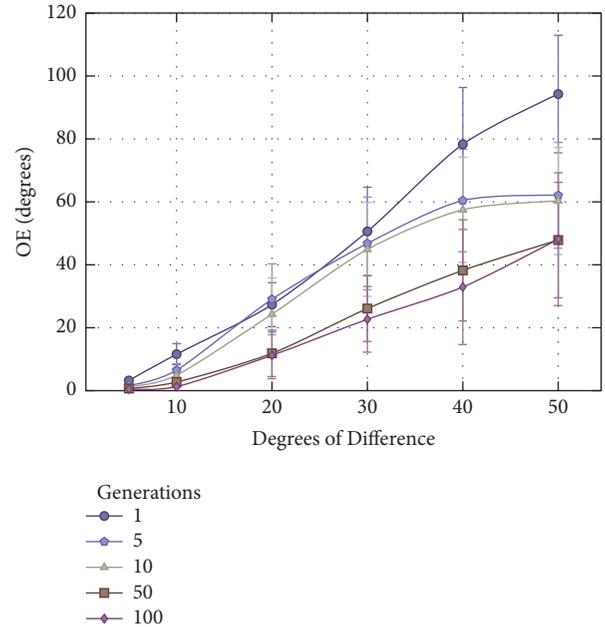


FIGURE 15: Results of orientation estimation of the proposed algorithm in noncontinuous video sequences in terms of OE. The error bars represent the standard deviation of measurements for 400 frames in 30 separate sample runs.

refined through generations of the evolutionary correlation filtering algorithm.

**4.5. Performance Evaluation in Noncontinuous Video Sequences with Real Scenes.** We present the evaluation

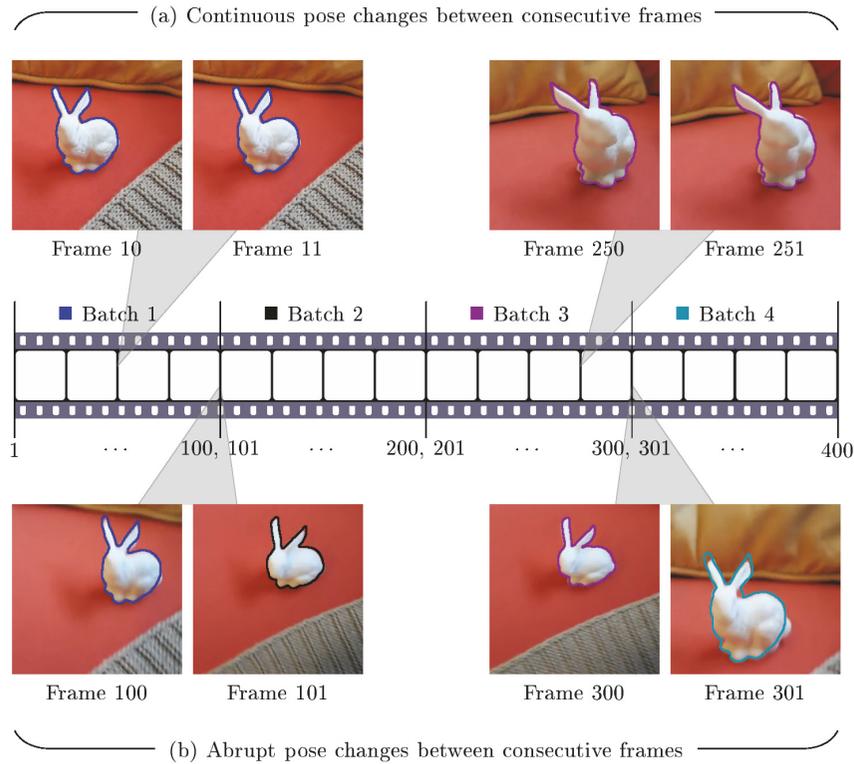


FIGURE 16: Results of pose estimation obtained with the proposed evolutionary correlation filtering approach in a real video sequence. (a) The algorithm evolves into optimal estimates in the subsequent frames. (b) The algorithm supports abrupt pose changes in noncontinuous target displacement. (Video 1, MPEG, 1.3 MB).

performance of the proposed algorithm for pose estimation in real scenes. In this experiment, we processed a video sequence of 400 frames with  $256 \times 256$  pixels and RGB color channels. The observed video presents a printed object based on the 3D model from [38]. The frame sequence contains the target with unknown pose parameters of location, orientation, and scaling. Video 1 shows the performance of the proposed algorithm for real images. The proposed algorithm successfully estimates the pose of the object despite the presence of challenging factors. In Video 1, the observation point presents rough displacements in the scene, yielding a different appearance of the object through consecutive frames. Also, in this experiment, we used four different batches, each one corresponding to a set of 100 frames. We scrambled the order of the batches for obtaining abrupt pose changes of the target. Figure 16 illustrates the performance of the evolutionary correlation filtering algorithm for the tested video sequence. From Figure 16(a), we can see that the input frames processed by the proposed algorithm evolve into optimal pose estimates in subsequent frames for the same batch. Figure 16(b) shows that the algorithm supports abrupt pose variations when the current batch suddenly changes. The algorithm provides robustness in noncontinuous target displacements given by the evolutionary approach in the correlation filtering implementation.

## 5. Conclusions

In this paper, an evolutionary correlation filtering approach for pose estimation in noncontinuous video sequences was presented. The proposed algorithm exhibited that it is robust for 3D pose estimation and tracking of a target in challenging images conditions such as additive noise, cluttered background, and motion blur, caused for imprecise video capturing. The performance of the proposed algorithm was tested by processing video sequences containing noncontinuous segments and abrupt pose changes of the target. Accordingly, with the obtained results, the proposed algorithm was able to adapt the correlation filtering process using an evolutionary approach. This method was able to successfully refine the final pose estimation solution given by the correlator. For pose tracking, the algorithm used information of the pose estimation in previous frames by assuming that the object displacement within the scene appears continuously. Additionally, the evolutionary approach allowed dealing with a wide range of pose candidate solutions. Thus, it yields high diversity in the exploration of the landscape of solutions to find the best pose parameters. Hence, the evolutionary correlation filtering presented robustness under abrupt pose changes. Based on the obtained experimental results the proposed algorithm yielded a high accuracy of pose tracking for noncontinuous video sequences, given in terms

of orientation and location errors. The proposed algorithm also showed its versatility of the correlation filtering used for three-dimensional pose estimation applications. Thanks to the evolutionary approach, the filter bank adapts to the best (suboptimal or optimal in the best of cases) solution. Computer simulation results proved accurate results for real-world scenes in terms of pose estimation obtained with the proposed evolutionary correlation filtering approach.

### Data Availability

The data used to support the findings of this study are included within the article.

### Conflicts of Interest

The authors declare that they have no conflicts of interest.

### Acknowledgments

This research was supported by Consejo Nacional de Ciencia y Tecnología (CONACYT).

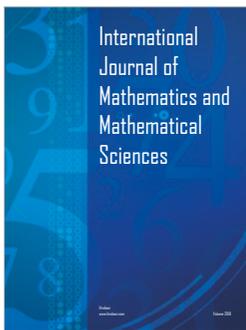
### Supplementary Materials

Video 1 (MPEG, 1.3 MB) shows the performance of the evolutionary correlation filtering for real images. The video sequence contains 400 frames with 256×256 pixels and 3 color channels RGB. In Video 1, it can be observed how the proposed algorithm successfully estimates the pose of the object despite the presence of challenging factors. The observation point presents rough displacements in the scene, yielding abrupt pose changes of the object through consecutive frames. (*Supplementary Materials*)

### References

- [1] J. Yu, J. Sun, S. Luo, and B. Duan, "Coupled multiview autoencoders with locality sensitivity for three-dimensional human pose estimation," *Journal of Electronic Imaging*, vol. 26, no. 5, 2017.
- [2] Qi Peng, Lifen Tu, Kaibing Zhang, and Sidong Zhong, "Automated 3D Scenes Reconstruction Using Multiple Stereo Pairs from Portable Four-Camera Photographic Measurement System," *International Journal of Optics*, vol. 2015, Article ID 471681, 9 pages, 2015.
- [3] Y. Yoshiyasu and N. Yamazaki, "Pose Space Surface Manipulation," *International Journal of Computer Games Technology*, vol. 2012, pp. 1–13, 2012.
- [4] Z. Liu and F. Cohen, "Synthesis and identification of three-dimensional faces from image(s) and three-dimensional generic models," *Journal of Electronic Imaging*, vol. 26, no. 06, p. 1, 2017.
- [5] J. Yu, J. Sun, Z. Song, S. Zheng, and B. Wei, "Monocular three-dimensional human pose estimation using local-topology preserved sparse retrieval," *Journal of Electronic Imaging*, vol. 26, no. 3, p. 033008, 2017.
- [6] Xiang Wu and Ning Wu, "Computationally Efficient Iterative Pose Estimation for Space Robot Based on Vision," *Journal of Robotics*, vol. 2013, Article ID 692838, 7 pages, 2013.
- [7] Dianyong Zhang, Zhenjiang Miao, Shengyong Chen, and Lili Wan, "Optimization and Soft Constraints for Human Shape and Pose Estimation Based on a 3D Morphable Model," *Mathematical Problems in Engineering*, vol. 2013, Article ID 715808, 8 pages, 2013.
- [8] Lei Zhang, Yanjie Wang, Honghai Sun, Zhijun Yao, and Shuwen He, "Robust Visual Correlation Tracking," *Mathematical Problems in Engineering*, vol. 2015, Article ID 238971, 13 pages, 2015.
- [9] B. V. K. V. Kumar, A. Mahalanobis, and R. D. Juday, *Correlation Pattern Recognition*, Cambridge University Press, 2005.
- [10] P. M. Aguilar-González and V. Kober, "Correlation Filters for Pattern Recognition Using a Noisy Reference," in *Progress in Pattern Recognition, Image Analysis and Applications*, vol. 5197 of *Lecture Notes in Computer Science*, pp. 38–45, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.
- [11] Y. K. Seong and T. S. Choi, "Optimal-trade-off filters for noise robustness, peak sharpness, and light efficiency in nonoverlapping background noise," *Optical Engineering*, vol. 39, no. 2, pp. 472–477, 2000.
- [12] A. Ruchay, V. Kober, and J. A. Gonzalez-Fraga, "Reliable Recognition of Partially Occluded Objects with Correlation Filters," *Mathematical Problems in Engineering*, vol. 2018, Article ID 8284123, 8 pages, 2018.
- [13] K. Picos, V. H. Diaz-Ramirez, V. Kober, A. S. Montemayor, and J. J. Pantrigo, "Accurate three-dimensional pose recognition from monocular images using template matched filtering," *Optical Engineering*, vol. 55, no. 6, 2016.
- [14] J. Lee, R. Sandhu, and A. Tannenbaum, "Particle filters and occlusion handling for rigid 2D-3D pose tracking," *Computer Vision and Image Understanding*, vol. 117, no. 8, pp. 922–933, 2013.
- [15] K. Picos, V. H. Diaz-Ramirez, A. S. Montemayor, J. J. Pantrigo, and V. Kober, "Three-dimensional pose tracking by image correlation and particle filtering," *Optical Engineering*, vol. 57, no. 07, p. 1, 2018.
- [16] V. H. Diaz-Ramirez, A. Cuevas, V. Kober, L. Trujillo, and A. Awwal, "Pattern recognition with composite correlation filters designed with multi-objective combinatorial optimization," *Optics Communications*, vol. 338, pp. 77–89, 2015.
- [17] O. Montiel and F. J. Díaz Delgadillo, "Reducing the Size of Combinatorial Optimization Problems Using the Operator Vaccine by Fuzzy Selector with Adaptive Heuristics," *Mathematical Problems in Engineering*, vol. 2015, 2015.
- [18] L. Blum, "Computing over the reals: where Turing meets Newton," *Notices*, vol. 51, no. 9, pp. 1024–1034, 2004.
- [19] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, San Francisco, Calif, USA, 1979.
- [20] A. Martínez-Vargas, J. Domínguez-Guerrero, Á. G. Andrade, R. Sepúlveda, and O. Montiel-Ross, "Application of NSGA-II algorithm to the spectrum assignment problem in spectrum sharing networks," *Applied Soft Computing*, vol. 39, pp. 188–198, 2016.
- [21] V. H. Diaz-Ramirez, L. Trujillo, and S. Pinto-Fernandez, "Advances in object recognition systems," in *Advances in Adaptive Composite Filters for Object Recognition*, I. Kypraios, Ed., pp. 91–110, 2012.
- [22] L. Altenberg, "Evolutionary Computation," *Encyclopedia of Evolutionary Biology*, pp. 40–47, 2016.

- [23] O. Kramer, "Genetic Algorithms," in *Genetic Algorithm Essentials*, vol. 679 of *Studies in Computational Intelligence*, pp. 11–19, Springer International Publishing, Cham, 2017.
- [24] O. Montiel, R. Sepúlveda, and U. Orozco-Rosas, "Optimal path planning generation for mobile robots using parallel evolutionary artificial potential field," *Journal of Intelligent & Robotic Systems*, vol. 79, no. 2, pp. 237–257, 2014.
- [25] U. Orozco-Rosas, O. Montiel, and R. Sepúlveda, "High-performance navigation system for mobile robots," in *High Performance Programming for Soft Computing*, O. Montiel and R. Sepúlveda, Eds., vol. ch. 12, pp. 258–281, CRC Press, Boca Raton, FL, USA, 2014.
- [26] B. Javidi and J. Wang, "Design of filters to detect a noisy target in nonoverlapping background noise," *Journal of the Optical Society of America A: Optics, Image Science & Vision*, vol. 11, no. 10, p. 2604, 1994.
- [27] V. Kober and J. Campos, "Accuracy of location measurement of a noisy target in a nonoverlapping background," *Journal of the Optical Society of America A: Optics and Image Science, and Vision*, vol. 13, no. 8, pp. 1653–1666, 1996.
- [28] V. H. Diaz-Ramirez, K. Picos, and V. Kober, "Target tracking in nonuniform illumination conditions using locally adaptive correlation filters," *Optics Communications*, vol. 323, pp. 32–43, 2014.
- [29] M. Mitchell, "Genetic algorithms," in *1992 lectures in complex systems (Santa Fe, NM, 1992)*, Santa Fe Inst. Stud. Sci. Complexity Lectures, V, pp. 3–87, Addison-Wesley, Reading, MA, 1993.
- [30] O. Castro, V. H. Diaz-Ramirez, A. Diaz-Ramirez, and V. Kober, "Improvement of pattern recognition with a heuristic design of correlation filters," in *Proceedings of the SPIE Optical Engineering + Applications*, p. 744217, San Diego, CA.
- [31] S. D. Dao, K. Abhary, and R. Marian, "A bibliometric analysis of Genetic Algorithms throughout the history," *Computers & Industrial Engineering*, vol. 110, pp. 395–403, 2017.
- [32] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*, University of Michigan Press, Oxford, UK, 1975.
- [33] D. B. Fogel, *Evolutionary Computation: Principles and Practice for Signal Processing*, SPIE, Bellingham, Wash, USA, 2000.
- [34] T. Back, U. Hammel, and H.-P. Schwefel, "Evolutionary computation: Comments on the history and current state," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 3–17, 1997.
- [35] C.-H. Wu, G.-H. Tzeng, Y.-J. Goo, and W.-C. Fang, "A real-valued genetic algorithm to optimize the parameters of support vector machine for predicting bankruptcy," *Expert Systems with Applications*, vol. 32, no. 2, pp. 397–408, 2007.
- [36] D. B. Fogel, "Introduction to simulated evolutionary optimization," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 5, no. 1, pp. 3–14, 1994.
- [37] U. Orozco-Rosas, O. Montiel, and R. Sepúlveda, "Parallel Evolutionary Artificial Potential Field for Path Planning—An Implementation on GPU," in *Design of Intelligent Systems Based on Fuzzy Logic, Neural Networks and Nature-Inspired Optimization*, vol. 601 of *Studies in Computational Intelligence*, pp. 319–332, Springer International Publishing, Cham, 2015.
- [38] G. Turk and M. Levoy, "Stanford bunny 3D digital model," <http://graphics.stanford.edu/data/3Dscanrep>.
- [39] B. T. Phong, "Illumination for computer generated pictures," *Communications of the ACM*, vol. 18, no. 6, pp. 311–317.



**Hindawi**

Submit your manuscripts at  
[www.hindawi.com](http://www.hindawi.com)

