Evolutionary Optimization Applied for Fine-Tuning Parameter Estimation in Optical Flow-based Methods

Danillo R. Pereira, Francisco Silva
FIPP
UNOESTE - Brazil
danilopereira,chico@unoeste.br

João P. Papa, Marco A. Piteri, Almir O. Artero
Dep. of Computing
Unesp - Brazil
papa@fc.unesp.br, {piteri,almir}@fct.unesp.br

Jose Delpiano
Faculty of Engineering
University of the Andes - Chile,
jd@miuandes.cl

Abstract—Optical flow methods are accurate algorithms for estimating the displacement and velocity fields of objects in a wide variety of applications, being their performance dependent on the configuration of a set of parameters. Since there is a lack of research that aims to automatically tune such parameters, in this work we have proposed an evolutionary-based framework for such task, thus introducing three techniques for such purpose: Particle Swarm Optimization, Harmony Search and Social-Spider Optimization. The proposed framework has been compared against the well-known Large Displacement Optical Flow (LDOF) approach, obtaining the best results in four out eight image sequences provided by a public dataset. Additionally, the proposed framework can be used with any other optimization technique.

Keywords—Social-Spider Optimization, Optical Flow, Evolutionary Optimization Methods

I. INTRODUCTION

Different Optical Flow (OF) techniques have been actively pursued in the last decades, since to estimate the displacement and the velocity of moving objects is a theme of great interest for general applications. New optical flow methods were developed in the last years, leading to some advancements in this area, as well as better matching results. However, the majority of these approaches requires as input a set of parameters for tuning their performance. Generally, these parameters are set empirically, which may be insurmountable for large image-based sequences. In addition, a manual tuning of the models requires a substantial human effort, being also more prone to errors. This can also limit the understanding of how optical flow methods does work in practice.

In the relative literature, most works do not consider the optimal selection of their set of parameters. They rather fine-tune the parameters manually, usually aiming to minimize some error rate metric. An exception can be found in the work of Salmen et al. [1], where the authors used a multi-objective (MO) evolutionary-based approach named “Covariance Matrix Adaption” to obtain more accurate optical flow algorithms. Very recently, Delpiano et al. [2] employed MO Genetic Algorithms for the same context. Although MO optimization algorithms can address multiple fitness functions, they might require more computational burden than mono-objective optimization algorithms, since the set of optimal solutions need to be constrained to the well-known Pareto Optimality.

In this context, it is possible to observe a relative lack of investigation of evolutionary optimization techniques applied for parameter tuning. The task of choosing proper values for OF methods can represent a large combinatorial problem that can be well-fitted by evolutionary algorithms, since such sort of techniques have been proposed in order to avoid traps from local optima. Therefore, this work aims to overcome this lack introducing three methods to the problem of mono-objective parameter tuning in the context of optical flow estimation: Particle Swarm Optimization (PSO) [3], Harmony Search (HS) [4] and Social-Spider Optimization (SSO) [5].

PSO is a method that simulates a swarm intelligence that finds a solution in a search space based on the dynamics of social behavior [3]. Each possible solution of the problem is modeled as a particle swarm that imitates its neighbours based on an objective function. Later on, HS was proposed and inspired from the improvisation process of musicians, more precisely jazz musicians [4]. This approach has been highlighted in the last decade due to the fast convergence and reduced number of calculations. Finally, based on the social dynamics of spiders, Cuevas et al. [5] proposed the Social-Spider Optimization, which considers both male and female spiders as well as their cooperative behaviour for solving optimization tasks. Such technique has demonstrated very promising results, being also so efficient as some state-of-the-art evolutionary-based approaches. The three evolutionary optimization techniques can be seen as heuristic approach that is based on a stochastic process.

Therefore, this paper aims to evaluate PSO, HS and SSO in the context of optical flow parameter estimation, since such techniques have never been employed for such purpose so far. The main contributions of this paper are two-fold: (i) to introduce PSO, HS and SSO for OF parameter estimation, as well as (ii) to compare them in such context. The remainder of this paper is organized as follows. Section II presents a brief theory background regarding optical flow and describes the evolutionary methods used in this work. Section III outlines the methodology employed in this work. Experiments are described in Section IV, and the conclusions are stated in Section V.

II. BACKGROUND THEORY

A. Optical Flow

Optical flow is a vector field representing “the distribution of apparent velocities of movement of brightness patterns in an image” [6]. The idea contains two basic assumptions: the “grey value constancy” and the “smooth flow of the intensity values”
between two successive images. Some articles still maintain
the grey value constancy (as an example, see [7]), while other
works report the necessity to loosen this assumption [8].

The OF constraint, given in Equation 1, is derived from
the “grey value constancy” assumption. It relates the spatial
and temporal derivatives of a 2D image \( f = f(x, y, t) \) and
the OF vector \( \mathbf{w} \). It has a strong analogy with mass conservation
in fluid mechanics, shown in Equation 2, where \( \mathbf{w} \) is the fluid
speed and \( \rho \) is the fluid density. As fluid mass, image intensity
is often supposed to remain constant under deformation and
motion. However, Equation 1 and Equation 2 would only be
equivalent exactly when \( \nabla \mathbf{w} = 0 \). This condition matches the
smooth flow assumption, that is considered when regularizing
the flow field:

\[
\frac{\partial f}{\partial t} + \mathbf{w} \nabla f = 0, \quad (1)
\]

\[
\frac{\partial \rho}{\partial t} + \nabla (\rho \mathbf{w}) = 0. \quad (2)
\]

The early work presented in [6] stated the need for an
extra constraint to compute the optical flow field from an
image sequence, and proposed one ad-hoc constraint based on
the assumption of flow smoothness. Another early research
work [9] proposed the consideration of the OF equation for
several neighbouring pixels in order to avoid the need for an
extra constraint. More than ten years later, Barron et al. [10]
gave a comparison of several OF methods, mainly with respect
to their average angular error (AAE) when applied to some
image sequences. The tests in that study showed that the
method in [9] was one of the most reliable methods at the
moment. Several image data sets have been compiled recently
for more precise evaluation and comparison of OF methods
[11] [12]. Many problems of the original methods have been
overcome and the accuracy of the OF methods on the top of
the rankings has grown continuously. Several researchers
have tried to preserve the discontinuity of natural motion fields
[13], overcoming the original assumption of OF smoothness
in [6]. After the comparison in [10], there have been further
attempts to compare different methods. For example, in [14]
the authors show a trade-off between computation time of an
algorithm and angular error obtained, given a fixed density.
They use operation curves of the OF algorithms to compare
them. It is interesting to read a time comparison among OF
algorithms given by [15], because they provide a picture of
the computational load of OF algorithms. More recently, a
large group of researchers presented a series of real image sequences
with ground truth OF obtained by tracking hidden fluorescent
textures [11]. The authors also suggest a method for evaluation
of OF algorithms. As a partial conclusion for OF, the accuracy
of OF methods has improved and allows for usage in some
common computer vision applications.

Large Displacement Optical Flow: Let \( \mathbf{w} := (u, v)^T \)
be the optical flow for a pair of consecutive frames \( I_1, I_2 \),
being such frame pre-smoothed using a Gaussian filter with
parameter \( \sigma \). The large displacement optical flow (LDOF)
method proposed by Brox and Malik [16] solves the energy
functional given by:

\[
E(\mathbf{w}) = E_{\text{color}}(\mathbf{w}) + \gamma E_{\text{gradient}}(\mathbf{w})
+ \alpha E_{\text{smooth}}(\mathbf{w}) + \beta E_{\text{match}}(\mathbf{w}, \mathbf{w}_1)
+ E_{\text{desc}}(\mathbf{w}_1), \quad (3)
\]

where the term \( E_{\text{color}} \) represents the common assumption of
grey value or color constancy; \( E_{\text{gradient}} \) represents gradient
constancy, which is invariant to an uniform illumination
change; \( E_{\text{smooth}} \) enforces regularity of the resulting optical
flow; \( E_{\text{match}} \) an energy related to point correspondences; mini-
imization of \( E_{\text{desc}} \) assures descriptor matching. The quantity \( \mathbf{w}_1 \)
is an auxiliary variable which allows to integrate descriptor
matching into a continuous approach. The implementation
available for LDOF\(^1\) allows the user to change the parameters
\( \sigma, \alpha, \beta, \gamma \). This set of parameters influence significantly on the
accuracy (consequently on the error metrics) and at runtime.

B. Particle Swarm Optimization

The PSO can be seen as a search algorithm based on
stochastic processes [3], where the learning of social behavior
allows each possible solution (particle) “fly” within that space
(swarm) looking for other particles that have the best features,
and thus minimizing or maximizing the objective function.

Each particle has a memory for storing its best local
solution (local maxima or minima) and the global best solution
(global maximum or minimum). Taking this into account,
each particle has the ability to imitate others that provide
the best positions in the swarm. This mechanism can be
summarized in three principles: (i) evaluation, (ii) comparison
and (iii) imitation. Each particle can evaluate others within
your neighborhood through some objective function, it can
compare with your own value, and finally decide whether it is
a good choice to imitate it.

The swarm is modeled as a multidimensional space \( \mathbb{R}^n \),
where each particle \( x_i = (p_i, v_i) \in \mathbb{R}^n \) has two main features:
(i) position \( p_i \) and (ii) velocity \( v_i \). The best local \( \tilde{p}_i \) and global
\( \bar{s} \) solution (position in the swarm) are also known. After setting
the size of the swarm (the number of particles), each particle
is initialized with random values for velocity and position.
Each particle is then evaluated with respect to some objective
function and its local maxima/minima is updated. The global
maximum/minimum value is updated with the particle that
reached the best position in the swarm. This process is repeated
until some convergence criterion. The position and velocity of
the particle \( x_i \) at time step \( t + 1 \) are updated by Equations 4
and 5, respectively:

\[
p_i(t + 1) = p_i(t) + v_i(t),
\]

\[
v_i(t + 1) = \Psi v_i(t) + c_1 r_1(\tilde{p}_i(t) - x_i(t)) + c_2 r_2(\bar{s} - p_i(t)) \quad (5)
\]

where \( \Psi \) is the inertia force that controls the interaction power
between particles, and \( r_1, r_2 \in [0, 1] \) are random variables
that give the idea of stochasticity for the PSO method. The constants
\( c_1 \) and \( c_2 \) are also used to guide the particles (input
parameters for the algorithm).

C. Harmony Search

Harmony Search (HS) is a meta-heuristic method that is
based on the improvisation process of musicians searching for
a good harmony [4]. This method has a theoretical stochastic

\(^1\)http://lmb.informatik.uni-freiburg.de/resources/software.php, accessed on
April 7th, 2014.
background. Each iteration of this approach generates a new harmony vector \( h_{\text{new}} = (h_{1\text{new}}, h_{2\text{new}}, ..., h_{n\text{new}}) \) based on memory considerations, small adjustments, and randomization, i.e., music improvisation (\( n \) is the number of decision variables).

The idea of the memorization step is to model the process of creating songs, in which the musician can use his/her memories of good musical notes to create a new song. This process is modeled by the Harmony Memory Considering Rate (HMCR) parameter; mathematically:

\[
h_{j\text{new}}^i \leftarrow \begin{cases} h_{j\text{new}}^i & \text{with probability HMCR} \\ S_j & \text{with probability (1-HMCR)}, \end{cases}
\]

where \( m \) and \( S_j \) are the number of harmonies and the set of ranges for each decision variable \( j \), respectively. Therefore, \( \text{HMCR} \in [0,1] \) is the probability of choosing one value from the historic values stored in the harmony memory, and \( (1-\text{HMCR}) \) is the probability of randomly choosing one feasible value. Further, every component \( j \) of the new harmony vector \( h_{\text{new}} \) is examined to determine whether it should be pitch-adjusted, which is controlled by the Pitch Adjusting Rate (PAR) variable:

\[
h_{j\text{new}}^i \leftarrow \begin{cases} \text{Yes}, & \text{with probability PAR} \\ \text{No}, & \text{with probability (1-PAR)}. \end{cases}
\]

The pitch adjustment for each instrument is often used to improve solutions and to avoid local optima. This mechanism concerns shifting the neighboring values of some decision variable in the harmony. As such, if the pitch adjustment decision for the decision variable \( h_{j\text{new}}^i \) is Yes, then \( h_{j\text{new}}^i \) is replaced as follows:

\[
h_{j\text{new}}^i \leftarrow h_{j\text{new}}^i + \psi_i b,
\]

where \( b \) is an arbitrary distance (bandwidth) for the continuous design variable, and \( \psi_i \in [0,1] \) is an ad-hoc parameter.

In the recent years, several researches have focused on developing variants from the traditional HS [17], [18]. Some of them propose ways to set the HS parameters dynamically, and others propose new improvisation schemes. In our implementation, we employed the Novel Global Harmony Search (NGHS) [18], since this variation has shown to be the most accurate. The NGHS depends on a single parameter \( P \) that denotes the probability of occurrence of an improvisation during a new particle’s creation.

D. Social-Spider Optimization

Social-Spider Optimization is based on the cooperative behavior of social-spiders [5], and it takes into account two genders of search spiders: males and females. Depending on the gender, each agent is conducted by a set of different operators emulating a cooperative behavior in a colony. The search space is assumed as a communal web and a spider’s position represents an optimal (near optimal) solution. An interesting characteristic of social-spiders is the female-biased population. The number of male spiders hardly reaches 30% of the total colony members. The number of females \( N_f \) is randomly selected within a range of 65-90% of the entire population \( N \), being calculated as follows:

\[ N_f = [(0.9 - \xi 0.255)N], \]

where \( \xi \) is a random number between [0,1]. The number of male spiders \( N_m \) is given by:

\[ N_m = N - N_f. \]

Each spider \( i \) receives a weight \( \phi_i \) according to the fitness value of the solution:

\[ \phi_i = \frac{\text{fitness}_i - \text{worst}}{\text{best} - \text{worst}}, \]

where \( \text{fitness}_i \) is the fitness value obtained by the evaluation of the \( i \)th spider’s position \( i = 1, 2, ..., N \). The \( \text{worst} \) and \( \text{best} \) mean the worst fitness value and best fitness value of the entire population, respectively.

The communal web is used as a mechanism to transmit information among the colony members. The information is encoded as small vibrations and depends on the weight and distance of the spider which has generated them:

\[ V_i,j = \phi_i e^{-d_{i,j}}, \]

where \( d_{i,j} \) is the Euclidean Distance between the spider \( i \) and \( j \). We can consider three special relationships:

- Vibrations \( V_{i,c} \) are perceived by the spider \( i \) as a result of the information transmitted by the member \( c \) who is the nearest member to \( i \), and possesses a higher weight \( \phi_c > \phi_i \);
- The vibrations \( V_{i,b} \) perceived by the spider \( i \) as a result of information transmitted by the best spider holding the best weight of the entire population;
- The vibrations \( V_{i,f} \) perceived by the spider \( i \) as a result of the information transmitted by the nearest female \( f \).

Social-spiders perform cooperative interaction over other colony members depending on the gender. In order to emulate the cooperative behavior of the female spider, a new operator is defined in Equation 13. The movement of attraction or repulsion \( \varphi_i \) of a female spider \( i \) at time step \( t + 1 \) is developed over other spiders according to their vibrations, which are emitted over the communal web:

\[
\varphi_i(t+1) = \begin{cases} \varphi_i(t) + \alpha \ast V_{i,c} \ast (s_c - \varphi_i(t)) + \beta \ast V_{i,b} \ast (s_b - \varphi_i(t)) + \gamma \ast (\text{rand} - \frac{1}{2}) & \text{if } r_m < PF; \\ \varphi_i(t) - \alpha \ast V_{i,c} \ast (s_c - \varphi_i(t)) - \beta \ast V_{i,b} \ast (s_b - \varphi_i(t)) + \gamma \ast (\text{rand} - \frac{1}{2}) & \text{if } r_m \geq PF; \end{cases}
\]

where \( r_m, \alpha, \beta, \gamma \) and \( \text{rand} \) are uniform random numbers between [0,1], and \( s_c \) and \( s_b \) represent the nearest member to \( i \) that holds a higher weight and the best spider of the entire population, respectively.
The male spider population is divided into two classes: dominant and non-dominant. The dominant class spider has better fitness in comparison to non-dominant, and they are attracted to the closest female spider in the communal web. In the other hand, non-dominant male spiders tend to concentrate in the center of the male population as a strategy to take advantage of resources that are wasted by dominant males. The movement of male spiders is given by:

$$\delta_i(t+1) = \begin{cases} 
\delta_i(t) + \alpha * V_i \ast (s_f - \delta_i(t)) + \gamma * (\text{rand} - \frac{1}{2}) & \text{if } \phi_{N_f+i} > \phi^*; \\
\delta_i(t) + \alpha * \left( \frac{\sum_{h=1}^{N_f} f_h(t) \ast \phi_{N_f+h}}{\sum_{h=1}^{N_f} \phi_{N_f+h}} \right) & \text{if } \phi_{N_f+i} \leq \phi^*. 
\end{cases} \tag{14}$$

where $s_f$ represents the nearest female spider to the male spider $i$ and $\phi$ is the median weight of male spider population. Thus, the reader can observe we have distinct movement equations for male and female spiders. Mating is performed by dominant males and female members in a social-spider colony. Considering $r$ (calculated by Equation 15) as being the radius, when a dominant male spider locates female members inside $r$, it mates, forming a new brood:

$$r = \frac{\sum_{j=1}^{n} l_{high}^j - l_{low}^j}{2n} \cdot \tag{15}$$

where $n$ is the dimension of the problem, and $l_{high}^j$ and $l_{low}^j$ are the upper and lower bounds, respectively. Once the new spider is formed, it is compared to the worst spider of the colony. If the new spider is better, the worst spider is replaced by the new one.

### III. Methodology

This section describes the experimental setup as well as the results concerning the optimization methods using image sequences and their respective ground truths from the Middlebury\(^2\) dataset [11], which has been frequently used for the evaluation of different OF methods [11], [16]. The Middlebury dataset contains 8 synthetic and laboratory sequences with a dense ground truth, which can be observed in Figure 1.

We have employed the LDOF (Section II-A) together with our implementation of SSO, PSO and HS evolutionary optimization approaches. In order to compare the optimization methods, we computed the mean of End Point Error (EPE) [19] and Average Angular Error (AAE) [10] obtained by 5 executions for each of them. Such quantitative evaluation approaches are commonly used by the optical flow estimation community. As we are employing a single-objective optimization, we chose to optimize the value of EPE (we believe this approach is more feasible to be employed here).

The proposed methodology can be divided in two rounds, as depicted in Figure 2. In the first round, for each image sequence we used different parameters obtained by each of the evolutionary optimization methods, unlike the methodology used by LDOF baseline [16], that employed a single parameter set for all image sequences. In the second round, we executed the proposed optimization framework over LDOF for all eight sequences using the set of parameters that achieved the best results for each evolutionary optimization method obtained in the first round, being this set of parameters the one that has achieved the best overall performance (average EPE over all eight sequences).

The parameters used for each evolutionary optimization algorithm are displayed in Table I. Notice for all optimization algorithms we used 200 agents and 200 iterations. It is important to shed light over that such values have been set based on previous experiments [5], [20].

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSO</td>
<td>$c_1 = c_2 = 2.0$ and $\psi = 0.9$</td>
</tr>
<tr>
<td>SSO</td>
<td>$P^F = 0.5$</td>
</tr>
<tr>
<td>HS</td>
<td>$P = 0.1$</td>
</tr>
</tbody>
</table>

### IV. Experiments

This section presents the results obtained by PSO, HS and SSO for optical flow parameter optimization purposes. We would like to stress we did not consider the runtime, since our goal is to minimize the EPE metric. Furthermore, the parameters to be optimized have a strong influence on both EPE and runtime. Thus, in order to find a tradeoff between the execution time minimization and EPE, it is necessary to apply multi-objective optimization methods. Additionally, we did not use statistical methods for ranking the tests, since their standard deviations are too small for that.

In regard to the first round of experiments, Table II shows the results obtained applying PSO, HS and SSO using a different set of parameters for each image sequence. In regard to EPE measure, it is possible to observe that SSO achieved the best results in four out eight Middlebury sequences, and PSO obtained the remaining top four best values. In addition, HS did not achieve the best result in any of the tests, and obtained the second best performance in only one of the sequences (Grove2). If one consider the baseline (Table III), the proposed approaches have achieved better results in five sequences (Dimetrodon, Hydrangea, RubberWhale, Urban2 and Venus). However, it is important to point out we have used a different set of parameters for each image sequence, while the baseline used a single set of parameters for all sequences.

Therefore, in order to make a fair comparison against with the baseline, we performed a second round of experiments, so that we used a unique set of parameters for all image sequences. Table III compares the results obtained by the proposed approach using PSO, HS and SSH against with the LDOF baseline. Figure 3 shows the histograms of the results obtained for EPE and AAE using the best selected set of parameters. Figure 3(b) plots the comparison between the evolutionary optimization methods and the LDOF baseline [16].

In the second round of experiments, we achieved better results than LDOF baseline in four sequences (Dimetrodon, Hydrangea, Urban2 and Venus). Is worth noting that we did not perform a global optimization using the eight sequences, but an individual optimization in each sequence and posteriorly for
Fig. 1. Images from the Middlebury Dataset used in the experiments. From left to right: Dimetrodon, Grove2, Grove3, Hydrangea, Urban2, Urban3, RubberWhale and Venus.

The proposed framework consists of a bunch of evolutionary optimization techniques that model the problem of parameters’ tuning in optical flow-based environments, more precisely LDOF. In this work, we dealt with the problem of tuning parameters in optical flow-based environments, more precisely LDOF. We have employed PSO, SSO and HS to address this problem, being the former two optimizers in optical flow-based environments, more precisely LDOF.

Furthermore, we chose to optimize the EPE and not the AAE, as the LDOF baseline did. However, it is possible to observe the proposed evolutionary-based framework can outperform LDOF for some situations.

In this work, we dealt with the problem of tuning parameters in optical flow-based environments, more precisely LDOF. The proposed framework consists of a bunch of evolutionary optimization techniques that model the problem of parameters’ estimation as an optimization task. We have employed PSO, SSO and HS to address this problem, being the former two optimizers in optical flow-based environments, more precisely LDOF.

TABLE II. QUANTITATIVE ANALYSIS FOR OPTICAL FLOW OPTIMIZATION USING PSO, HS AND SSO. THE BEST EPE VALUES OBTAINED FOR EACH SEQUENCE ARE IN BOLD.

<table>
<thead>
<tr>
<th>Sequence</th>
<th>EPE</th>
<th>AAE</th>
<th>EPE</th>
<th>AAE</th>
<th>EPE</th>
<th>AAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimetrodon</td>
<td>0.08 ± 0.00</td>
<td>1.69 ± 0.00</td>
<td>0.09 ± 0.00</td>
<td>1.78 ± 0.00</td>
<td>0.09 ± 0.00</td>
<td>1.76 ± 0.04</td>
</tr>
<tr>
<td>Grove2</td>
<td>0.17 ± 0.00</td>
<td>2.07 ± 0.01</td>
<td>0.17 ± 0.00</td>
<td>2.08 ± 0.01</td>
<td>0.16 ± 0.00</td>
<td>2.01 ± 0.02</td>
</tr>
<tr>
<td>Grove3</td>
<td>0.67 ± 0.00</td>
<td>6.24 ± 0.04</td>
<td>0.68 ± 0.00</td>
<td>6.35 ± 0.06</td>
<td>0.66 ± 0.02</td>
<td>6.22 ± 0.09</td>
</tr>
<tr>
<td>Hydrangea</td>
<td>0.16 ± 0.00</td>
<td>2.03 ± 0.00</td>
<td>0.17 ± 0.00</td>
<td>2.08 ± 0.01</td>
<td>0.16 ± 0.00</td>
<td>2.01 ± 0.02</td>
</tr>
<tr>
<td>RubberWhale</td>
<td>0.10 ± 0.00</td>
<td>3.41 ± 0.04</td>
<td>0.11 ± 0.00</td>
<td>3.69 ± 0.04</td>
<td>0.10 ± 0.00</td>
<td>3.48 ± 0.02</td>
</tr>
<tr>
<td>Urban2</td>
<td>0.30 ± 0.00</td>
<td>2.62 ± 0.02</td>
<td>0.34 ± 0.00</td>
<td>2.85 ± 0.08</td>
<td>0.29 ± 0.00</td>
<td>2.54 ± 0.01</td>
</tr>
<tr>
<td>Urban3</td>
<td>0.48 ± 0.04</td>
<td>4.54 ± 0.27</td>
<td>0.63 ± 0.01</td>
<td>5.20 ± 0.28</td>
<td>0.44 ± 0.00</td>
<td>4.26 ± 0.01</td>
</tr>
<tr>
<td>Venus</td>
<td>0.28 ± 0.00</td>
<td>4.34 ± 0.01</td>
<td>0.34 ± 0.01</td>
<td>5.42 ± 0.40</td>
<td>0.29 ± 0.01</td>
<td>4.62 ± 0.43</td>
</tr>
</tbody>
</table>

TABLE III. THE BEST VALUES OF AAE OBTAINED BY THE LDOF BASELINE [10]. THE BEST AAE VALUES OBTAINED FOR EACH SEQUENCE ARE IN BOLD.

<table>
<thead>
<tr>
<th>Sequences</th>
<th>HS</th>
<th>PSO</th>
<th>SSO</th>
<th>LDOF baseline</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimetrodon</td>
<td>1.78</td>
<td>2.21</td>
<td>1.71</td>
<td>1.82</td>
</tr>
<tr>
<td>Grove2</td>
<td>3.12</td>
<td>3.04</td>
<td>2.87</td>
<td>2.09</td>
</tr>
<tr>
<td>Grove3</td>
<td>6.89</td>
<td>6.82</td>
<td>6.78</td>
<td>5.59</td>
</tr>
<tr>
<td>Hydrangea</td>
<td>2.73</td>
<td>3.04</td>
<td>2.12</td>
<td>2.32</td>
</tr>
<tr>
<td>RubberWhale</td>
<td>4.29</td>
<td>4.59</td>
<td>4.15</td>
<td>3.77</td>
</tr>
<tr>
<td>Urban2</td>
<td>3.03</td>
<td>2.94</td>
<td>2.70</td>
<td>3.99</td>
</tr>
<tr>
<td>Urban3</td>
<td>4.92</td>
<td>4.27</td>
<td>5.00</td>
<td>2.28</td>
</tr>
<tr>
<td>Venus</td>
<td>5.76</td>
<td>4.47</td>
<td>4.19</td>
<td>5.19</td>
</tr>
</tbody>
</table>

V. Conclusions

In this work, we dealt with the problem of tuning parameters in optical flow-based environments, more precisely LDOF. The proposed framework consists of a bunch of evolutionary optimization techniques that model the problem of parameters’ estimation as an optimization task. We have employed PSO, SSO and HS to address this problem, being the former two optimizers in optical flow-based environments, more precisely LDOF.
approaches the most accurate, which also outperformed the compared baseline (LDOF) in four out eight datasets. Given the fact that we did not use evolutionary methods in the global way, i.e., we aimed to optimize each sequence separately and then considering the best overall performance, the results were quite satisfactory.

The methodology described in this work provide an automatic way for parameters estimation, thus avoiding the use of brute-force methods or even manually set them. Notice the proposed approach is mono-objective, which means only one objective function is optimized, i.e., EPE in this work. In our tests, we disregard the runtime due to the fact that the chosen parameters greatly influence both the runtime as well as the error rate. In order to establish a trade-off among computational load and error rate minimization, one must use multi-objective optimization methods, being that one our future work.

ACKNOWLEDGMENTS

The authors are grateful to FAPESP grant #2013/20387-7, and CNPq grants #470571/2013-6 and #303182/2011-3.

REFERENCES


