Efficient Local Search in Imaging Optimization Problems with a Hybrid Evolutionary Algorithm

Igor MASLOV
Department of Computer Science, The City University of New York
New York, NY 10016, USA

ABSTRACT

The paper focuses on the efficiency of local search in a Hybrid evolutionary algorithm (HEA), with application to optimization problem frequently encountered in electronic imaging. Although HEA can significantly improve the overall performance of evolutionary search, the direct usage of methods of local optimization gives rise to a few performance problems including a noticeable additional cost of fitness evaluations attributed to local search; an excessive waste of computational resources on a particular chromosome that is later discarded by the global search; and a possible convergence to a sub-optimal solution when the actual distance from the global optimum is not sufficiently small for the local search to successfully descend to the minimum point. Computational performance of local search can be potentially improved by applying the following techniques: using direct search that can better accommodate shape irregularities of fitness function; adding randomness and periodically re-positioning the search, thus preventing it from converging to a sub-optimal point; creating a tree-like structure for each local neighborhood that keeps track of the explored search space; using cyclic vs. complete local search, thus cutting down the excessive cost attributed to discarded chromosomes; incorporating image response analysis and providing the algorithm with a means of deriving problem-specific knowledge that speeds up the solution. A two-phase cyclic local search is proposed that incorporates these techniques. A series of computational experiments with 2-dimensional grayscale images provide experimental support for the proposed approach and show that computational performance of local search in imaging optimization with HEA can be significantly improved.

Keywords: Hybrid Evolutionary Algorithm, Local Search, Stochastic Methods, Optimization Problem, Electronic Imaging.

1. INTRODUCTION

Electronic imaging is one of the indispensable areas of research and development in modern science and technology. Three important tasks frequently encountered in electronic imaging – image registration, object recognition, and content-based image retrieval – can be stated in the form of an inverse optimization problem. Essentially, all three seemingly different tasks attempt to map an image Img_1 of an object or a scene onto an image Img_0 of another (e.g., larger) scene. Since Img_1 or Img_0 are typically subject to some transformation A, e.g., perspective or affine, the actual problem reduces to finding a vector of parameters that define the transformation A. If F is a measure of the difference between the images during the mapping, then the optimization problem can be stated as follows. A feasible vector $\boldsymbol{V}^*$ of parameters has to be found that minimizes the difference F between the images Img_1 and Img_0, so that

$$F(\boldsymbol{V}) > F(\boldsymbol{V}^*), \text{ for all } \boldsymbol{V} \neq \boldsymbol{V}^*, \text{ or } \boldsymbol{V}^* = \mathbf{arg\ min}\,(F(A)). \quad (1)$$

The choice of the image difference measure F is dictated by the nature of the problem that one attempts to solve. Color palette and color distribution, image texture, and noticeable features like lines and corners, etc. can be used to define F. In the area of a grayscale imagery, measures based on the distribution of pixel gray values are frequently used to evaluate the difference between images. In this paper, the squared difference of the pixel values of the two images is utilized as a convenient measure of the difference F, due to its fairly high robustness and tolerance to noise:

$$F = \frac{\sum_{\Omega} (g_1(x', y') - g_0(x, y))^2}{\Omega^2}, \quad (2)$$

where $g_1(x',y')$ and $g_0(x,y)$ are the pixel gray values of the images Img_1 and Img_0, respectively, and Ω is the area of their overlap [1].

In real world imaging applications, the optimization problem stated in Equations (1) and (2) is by no means a trivial task. The image difference F is usually a highly nonlinear function that does not have a closed analytical form; it has to be computed numerically for every trial parameter vector $\boldsymbol{V}$. The images can represent complex multi-object scenes, cluttered, noisy, and significantly distorted by the transformation A, which makes the stated optimization problem a highly nonlinear, global, and multimodal, i.e., having an unknown number of local minima. Stochastic optimization methods have proved to be efficient approach to solving these types of problems [2]. In particular, evolutionary algorithms (EAs) are well suited for complex multi-dimensional and multi-modal, nonlinear real world problems [3] – [6].

Many researchers applied EAs to solving electronic imaging problems, including image registration [7] – [10], semantic scene interpretation [11] – [13], and feature selection [14] – [16]. Majority of the research utilizes different variants of the classical model of EAs, whereas a number of more efficient models have been developed over the last decade. One of the advanced models, a Hybrid evolutionary algorithm (HEA) incorporates traditional methods of local optimization into the global evolutionary search [17], [18].

This paper focuses on exploring some techniques that can potentially improve the computational performance of local search in HEA, with application to imaging optimization problem stated in Equations (1) and (2). Without loss of generality, 2-dimensional (2-D) grayscale images subject to a

partial affine transformation A are considered. The transformation A may include image translations DX and DY, rotation θ, and generally non-isotropic scaling factors SX and SY. A transformed vector $\boldsymbol{p}' = \{x',y'\}^{\mathrm{T}}$ of the original coordinates $\boldsymbol{p} = \{x,y\}^{\mathrm{T}}$ of a pixel $P \in Img_1$ can be computed as

$$\boldsymbol{p}' = A(\boldsymbol{p}) = SR\boldsymbol{p} + T, \tag{3}$$

where the matrices S, R, and T represent scaling, rotation, and translation, respectively. For the purpose of optimization, the transformation A in Equation (3) can also be written in the form of a combination of global transformation defining the location of the object in the scene (i.e., the rigid body transformation), and local transformation defining the deformation (i.e., the distortion) of the object, as follows:

$$A(\boldsymbol{p}) = LG_1\,\boldsymbol{p} + G_2, \tag{4}$$

where L is a local transformation caused by the matrix S of a non-isotropic scaling, G_1 is a part of a global transformation G resulted from the object rotation defined by the matrix R, and G_2 is a part of the global transformation G resulted from the translations along the x- and y-axis defined by the matrix T. As one can see from Equation (4), the transformation A is non-separable, i.e., the optimization search has to be conducted simultaneously in both global G and local L parameter spaces.

The paper is organized as follows. Section 2 discusses some problems associated with the use of local search in HEA, and proposes their potential solutions. Section 3 details the implementation of a particular two-phase cyclic model of local search. Section 4 outlines the concept of image local response and its use for improving the performance of local search. Section 5 presents the results of computational experiments with 2-D grayscale images utilizing the proposed techniques. Section 6 summarizes the findings presented in the paper.

2. PROBLEMS AND POTENTIAL SOLUTIONS FOR IMPROVING THE PERFORMANCE OF LOCAL SEARCH IN A HYBRID EVOLUTIONARY ALGORITHM

Evolutionary algorithms (EAs) belong to the category of global optimization methods. The algorithms are based on observations of physical processes that occur in nature; they attempt to mimic the real physical processes in the artificial computational environment [19] - [22]. Following the analogy with the living organisms, the algorithms use the term *chromosome* to denote a single candidate solution (i.e., the sought parameter vector $\boldsymbol{V}$) to an optimization problem. The algorithms work concurrently with the population of contending chromosomes. The quality of each chromosome is evaluated via its *fitness* corresponding to the objective function of the optimization problem. In this paper, the difference F between the images plays the role of the objective function of the optimization problem stated in Equations (1) and (2). The algorithms work in an iterative manner producing the next population (i.e., *generation*, in the evolutionary terminology) of chromosomes from the current population using various genetic operators, the most common of which are selection, crossover, and mutation.

Being fairly general and versatile, the classical random model of EAs frequently exhibits slow convergence in solving many practical problems. In order to improve the EAs performance, hybridization with other computational methods is needed. One of the efficient directions is a Hybrid evolutionary algorithm. The latter augments the classical model of EAs with traditional methods of local search and optimization [17], [18]. The idea behind HEA is relatively simple, albeit, very fruitful. The local neighborhoods near the best chromosomes found by the global evolutionary search are further explored in a greater detail, in an attempt to find better individuals. The idea of the local improvement of the solution finds its theoretical support, e.g., in the concept of Lamarckian evolution. The latter states that the individual's improvement achieved during the process of learning affects the individual's genetic structure, and can be mapped back to the genetic structure of the entire population [18].

Although the hybrid model can significantly improve the overall performance of EAs, the direct usage of local search in HEA gives rise to a few efficiency problems. The first problem is related to the noticeable overall cost of the additional evaluations of the fitness function F attributed to local search. In a typical imaging application, the lion's share of the computational cost of HEA belongs to the fitness evaluations. Every evaluation includes a transformation of the image according to Equation (3), a pixel-wise comparison of the images, and an evaluation of the difference F between the images according to Equation (2). The cost of the fitness evaluation grows as the size of the images increases.

Another problem is related to the fact that in general case, the function F is multimodal, i.e., it can have several local optimum solutions. The algorithm can not guarantee that any of the best chromosomes produced at any intermediate step of the global evolutionary search is located in the close proximity to the global minimum point. Consequently, the additional computational resources spent on exploring the chromosome's neighborhood will be wasted if the chromosome happened to belong to one of the local minima: the targeted chromosome can be replaced with some other, a better individual, at a later stage of the global search.

The third problem is associated with the actual distance from the closely investigated chromosome to the global optimum point. The distance might not be sufficiently small for the local search to successfully take off and descend to the optimum. The higher the dimensionality of the search space, the more likely the local search converges to a sub-optimal solution defined by those components of the parameter vector $\boldsymbol{V}$ that are located farther away from their optimal values.

This paper ivestigates the following means by which the aforementioned problems associated with local search in HEA can be potentially defied:

- Using direct, as opposed to gradient-based methods of local search,
- Adding randomness to the rigorously defined framework of a chosen local procedure,
- Engaging bookkeeping operations similar to those used in Tabu search,
- Engaging repetitive and partial, rather than complete local search,
- Providing problem-specific knowledge, in the form of response analysis.

Potential candidates for a local search procedure in HEA have to be evaluated on the basis of two main criteria: robustness and performance. Robustness shows how readily the algorithm can accommodate itself to various irregularities of the shape of the objective function. It also determines how close to the optimum solution the starting point of the search has to reside, in order for the algorithm to successfully descend to the optimum. Performance defines the quality of the final solution found with the algorithm, and computational cost at which this quality has been achieved.

There are two main categories of the traditional methods of local optimization that can be potentially utilized in HEA [23]:

- Gradient-based methods engaging the derivatives of the objective function,
- Direct search methods solely based on computing the values of the objective function.

Methods belonging to the first group are most commonly used in hybrid models, largely due to the fact that their properties and performance are well studied and understood by the mathematical community. Relatively simple Steepest descend algorithm, and more sophisticated Conjugate gradient method (CGM) are among the popular choices in this category. Since these methods heavily rely on the accurate computation of the function derivatives, they pose certain, often severe demands on the shape and smoothness of the objective function and its derivatives. Optimization problems encountered in real world imaging applications can not guarantee that these demands can be satisfied. Fitness function F describing the difference between images has inherently random nature, can be non-convex, and can exhibit shape singularities. Both the function and its derivatives have to be computed numerically, the latter using one of the known finite difference schemes.

Direct search methods have not received the same amount of attention as their gradient-based counterparts, largely due to theoretical problems encountered by mathematicians who attempted to lay a coherent theoretical foundation for these methods. Only recently, direct search has seen a revival of interest, in light of the latest theoretical findings [24]. Among the optimization practitioners, however, these methods remained popular, because of their relatively low computational demands, high reliability, and often satisfactory final results. One of the popular choices of direct search has been the Downhill simplex method (DSM) [23], [25]. The method is an iterative procedure that maintains a non-degenerate $(N+1)$ – dimensional simplex in the N – dimensional search space. The vertices of the simplex form a set of approximations to the parameter vector $\boldsymbol{V}$ and the objective function F. On each iteration, the algorithm attempts to move the worst vertex to a better location, so the value of the function at this vertex decreases. The simplex continuously changes its shape and moves in the parameter space, until it converges to a minimum point. The DSM search does not require additional function evaluations associated with computing partial derivatives of the function F, and is highly tolerant to shape irregularities of the latter, which makes it an ideal candidate for using in imaging optimization with HEA.

Occasionally, the DSM search can converge to a sub-optimal solution, in which case it is recommended to re-define the intermediate location of the simplex [26]. The addition of the local random search to the rigorously defined DSM procedure can help re-position the simplex at a specified intermediate step in the most advantageous manner, and thus prevent it from falling into a sub-optimal point. Randomness, however, introduces a disturbance into the otherwise ordered and unambiguous simplex movement. Some of the points of the search space that have already been visited by the simplex can be re-visited, as the result of the added randomness. Moreover, the simplex movement can even slip into a repetitive loop over the same region. Monitoring the search space and keeping track of the locations already visited by the simplex becomes an important issue that requires the employment of an additional mechanism. Such mechanism can be borrowed from one of the popular heuristic methods, Tabu search [27].

The next section of the paper gives a detailed description of a local search procedure that implements the directions outlined in this section, in order to defy the problems associated with the usage of local search in a Hybrid evolutionary algorithm, with application to imaging optimization problem stated in Equations (1) and (2).

3. TWO-PHASE CYCLIC LOCAL SEARCH IN A HYBRID EVOLUTIONARY ALGORITHM

In order to increase the computational efficiency of local search in HEA, a two-phase cyclic procedure based on DSM and random search was designed that works in the following way. The algorithm maintains a variable-length global list L_B of the fittest, i.e., having the lowest fitness values F chromosomes. The local neighborhood of each chromosome $\boldsymbol{V}_k$ in the list is explored in a greater detail, in an attempt to find a better individual. Firstly, a random search is performed with the radius of the locality $r_n(t)$ surrounding $\boldsymbol{V}_k$, and the number of candidate neighbors $n_n(t)$, computed as linear functions of time t. The qualified neighbors are selected using the uniform probability distribution inside the N-dimensional hyper-parallelepiped built around $\boldsymbol{V}_k$ as a center, where N is the dimensionality of the search space. Figure 1 illustrates the mechanism of random local search on a sample 2-D model, where n neighbors of the current chromosome $\boldsymbol{V}_k$ are drawn at random from the expanding neighborhood.

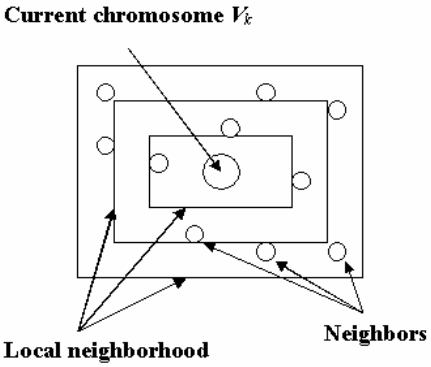

Figure 1. A sample 2-D model of the random search in the expanding neighborhood of the targeted chromosome $\boldsymbol{V}_k$.

Once a preset time threshold $t = T_{RND}$ has passed, the algorithm switches from the random search to the Downhill simplex method. The latter performs only a specified number n_c of complete cycles of the iterative local search each time it is

called by the global procedure. During one complete cycle, the initial simplex shrinks, and all its vertices change their location, thus decreasing their respective function values. After a second preset time threshold $t = T_{DSM}$ has passed, the local procedure switches from DSM back to the random search. At specified intervals $t = T_{SW}$, the simplex is re-positioned, as described below.

The search procedure keeps track of the neighbors selected during the random search by building a binary tree with the chromosome $\boldsymbol{V}_k$ as the initial root. Once the specified number of neighbors has been evaluated, the tree is rotated to the right. An individual $\boldsymbol{V}_j$ with the lowest value of the fitness function F_j replaces the current chromosome $\boldsymbol{V}_k$, and becomes a new root of the tree. When the procedure switches back to DSM, it chooses the chromosome $\boldsymbol{V}_j$ as the first vertex of the initial simplex. The other N vertices of the simplex are picked from the right brunch of the tree; these vertices have the next N lowest values of the fitness function. The selection technique ensures that the best-known simplex in the local neighborhood is formed as the starting point for the re-positioned DSM search. Figure 2 illustrates the mechanism that keeps track of the chromosomes located in the neighborhood of the current chromosome $\boldsymbol{V}_k$. The boxes represent the visited chromosomes in the neighborhood; they are labeled in accordance with their respective fitness values. The left-most leaf of the tree corresponds to a chromosome with the best value of F that is to replace the targeted chromosome $\boldsymbol{V}_k$ during the next cycle of the search.

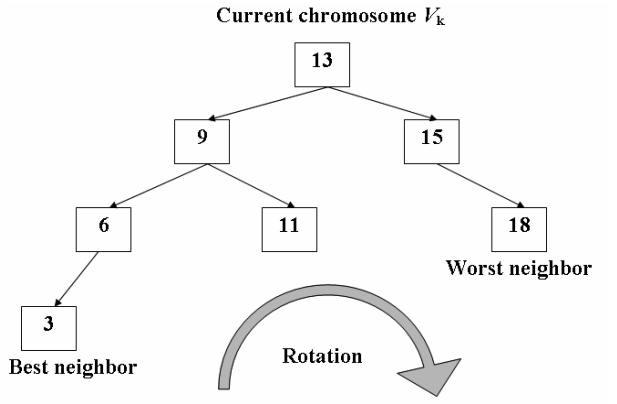

Figure 2. A sample binary tree of the neighborhood around the current chromosome $\boldsymbol{V}_k$.

The proposed algorithm of a two-phase cyclic local search that alternates the random and the DSM search is summarized in Figure 3. The procedure helps defy the efficiency problems related to the usage of local search in HEA described in the previous section of the paper. The cyclic implementation of the DSM search reduces the potential waste of the computational resources spent by DSM, in the case where $\boldsymbol{V}_k$ is replaced by the global search with a fitter candidate located outside the current neighborhood. In addition, alternating the random and the DSM search around the individual $\boldsymbol{V}_k$ re-defines the initial configuration of the simplex, thus helping prevent it from converging to a sub-optimal point. Further reduction of the overall cost of fitness evaluations associated with the local DSM search can be achieved using problem-specific knowledge, in the form of a peculiar image transformation, image local response, which is described in the next section of the paper.

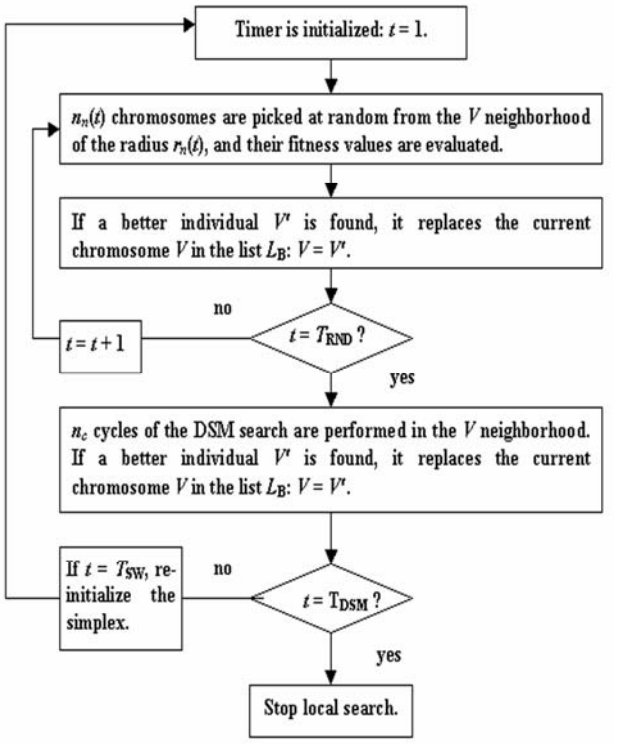

Figure 3. Flow chart of the two-phase cyclic local search procedure.

4. IMAGE LOCAL RESPONSE IN THE LOCAL DSM SEARCH

One of the strong points of EAs is their ability to deal with complex real world problems without any *a priori* provided problem-specific knowledge. On the flip side, however, the generality of EAs can make them less efficient, in comparison with *ad hoc* methods that extensively rely on problem-specific knowledge, when the latter is available. Luckily, the flexibility of EAs allows for the problem specifics to be incorporated into the general evolutionary framework. There are two principal ways of such incorporation:

- Providing the algorithm with the specific data about the problem,
- Providing the algorithm with a fairly general method that can autonomously obtain needed information and seamlessly integrate it into the evolutionary search.

Since the first approach leads to the apparent loss of the algorithm's generality, and in most cases requires human intervention, the second approach looks more desirable. This approach is used in the paper; it is rooted in the general ideas of sensitivity analysis, and implemented in the form of image local response.

The concept of image local response has a fairly simple underlying idea [28]. Since the statement of the optimization

problem in Equations (1) and (2) looks for a parameter vector $\boldsymbol{V}$ defining the unknown transformation A, it seems logical to explore the response of the image to this type of transformation. This task can be accomplished by mapping the transformed image *Img´* into its original version *Img*, with the small transformation vector $\boldsymbol{V_u}$. Thus, image local response R_P at a point P is defined as the value of the difference F between the transformed *Img´* and the original *Img* versions of the same image, where the transformation A_u at the point P is small, i.e. the components of the parameter vector $\boldsymbol{V_u}$ have sufficiently small unit values. It can be shown that there is no need to compute the response R_P over the entire image, since R_P rapidly decreases, as the distance from the point P increases. It suffices to compute the response R_P over a small pixel area ω_P near P, chosen as a response area. For convenience and without loss of generality, a square with a side r, considered as a radius of ω_P, can be chosen as the response area associated with the point P.

In order to build the response matrix M_R for the entire image *Img*, the local response R_P is computed for every point P of the image as follows:

1) Partial transformations A_n, $n = 1,\ldots,N$, are applied to the response area ω_P near the point P, such that each transformation A_n corresponds to a unit variation of one of the N parameters (e.g., *DX*, *DY*, θ, etc.).
2) For each partial transformation A_n, the difference F_n between the pixel values of the initial ω_P and the transformed ω_P' areas is computed according to Equation (2), where $\Omega = \omega_P$.
3) The value of the response R_P at the point P is computed as the averaged sum of all N differences F_n.

From the algorithmic point of view, the operation of computing local response is similar to the operation of image filtering. Very much like filtering, computing local response also produces a new output image Img_R, which serves as the graphical representation of the image response matrix M_R. Like filtering, computing local response is also a neighborhood operation, in which the value of any given pixel P in the output image Img_R is determined by the means of applying the described procedure to the values of the pixels in the local neighborhood ω_P of the pixel P in the input image (i.e., the original image *Img*).

It can be shown that image local response R_P has the following important properties:

1) As the radius r of the response area ω_P increases, the value of the response R_P rapidly decreases; it is bounded by the term $O(1/r)$.
2) For any two points P and Q that have similar pixel distributions over the corresponding response areas, the difference between their respective responses R_P and R_Q is fairly small.
3) When the pixel values in the response area near P significantly vary (e.g., near the object edges in the image), the value of the response R_P correspondingly increases.

The first property assures that the value of the local response R_P computed over a small area of the image can be used as a fairly good approximation to the response R_{Img} computed over the entire image. But the value of R_{Img}, in turn, corresponds to the value of the fitness function F computed for a small transformation A_u of the image *Img*. Since *Img* itself constitutes the optimum solution to the problem of mapping the transformed version *Img´* into its original version under the small transformation A_u, the local response R_P can be used as a fairly good approximation to the fitness function F computed at a point P in the vicinity of the optimum solution.

The second and the third properties indicate that the values of local response reflect the degree of smoothness of the fitness function F in the vicinity of the optimum solution. When F is fairly smooth, the distribution of the response values is relatively flat. On the contrary, any noticeable variation of F will cause the corresponding significant variation of the response values.

The aforementioned properties of image local response make it an efficient technique for reducing the overall computational cost of the fitness evaluations associated with the local DSM search. The drawback of the regular algorithm is that the step at which the simplex moves or changes its shape does not depend on the absolute values of the objective function (i.e., fitness function in HEA) at the vertices, nor does it depend on the differences between these values. The convergence of the simplex to the optimum solution is controlled only by the ranking order of its vertices, and by the values of four DSM coefficients - α_1, α_2, α_3, α_4 - that define reflection, expansion, contraction, and shrinkage of the simplex, respectively. The commonly used values of the coefficients are fixed, which makes the simplex move on a smooth landscape in the same way as on a rough landscape formed by the fitness function, as long as its vertices are aligned in the same ranking order.

The convergence of the simplex to the optimum can be adaptively controlled in such a way that the simplex will move at its regular rate on a relatively smooth fitness landscape, and will slow down on a rough landscape. Such control, however, would require additional fitness evaluations within the area covered by the simplex. The computational cost associated with the additional evaluations can outweigh the relative gain obtained from the adaptation. This is where image local response comes to the rescue: it provides a fairly good approximation to the fitness function F in the vicinity of the optimum solution, while evaluating F only over a small response area. Response R_P can sense the properties of the fitness landscape at a point P in the area covered by the DSM simplex, without the overhead of the expensive computation of the entire fitness function F at the specified point.

Following the above reasoning, image local response is utilized to control the length of the vector $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4\}$ of the DSM coefficients. When the fitness landscape in the locality is sufficiently smooth, the distribution of the response values is nearly flat, and the components of the vector $\boldsymbol{\alpha}$ take on their common fixed values. However, when fitness function significantly varies, the correspondingly changing response causes the contraction of the vector $\boldsymbol{\alpha}$. The new vector $\boldsymbol{\alpha'}$ of the coefficients can be found using the following expression [29]:

$$\boldsymbol{\alpha'} = \boldsymbol{T}(\boldsymbol{\alpha}) = \boldsymbol{D_{PQ}}\boldsymbol{\alpha}, \tag{5}$$

where $\boldsymbol{\alpha'} = \{\alpha_1', \alpha_2', \alpha_3', \alpha_4'\}$ is the vector of the modified DSM coefficients, and D_{PQ} is a diagonal matrix of the response coefficients C_{PQ} associated with image points P and Q, and computed as follows:

$$C_{PQ} = (1 - \ln(R_P / R_Q)) . \quad (6)$$

Experiments presented in the next section of the paper provide necessary support for using image local response as a means by which the computational cost of fitness evaluations during the local DSM search can be reduced.

5. COMPUTATIONAL EXPERIMENTS

In order to evaluate the efficiency of the proposed approach to local search in a Hybrid evolutionary algorithm, computational experiments were conducted on different test sets of 2-D grayscale images. Each test set includes images of an object and a scene. The object represents a part of the scene that undergoes partial affine transformation *A*, and has to be mapped back onto the scene.

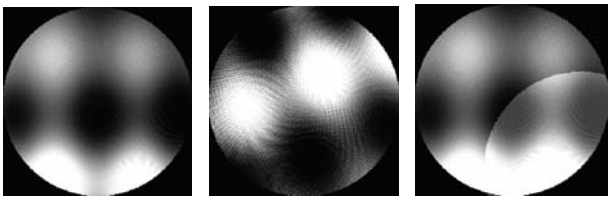

Figure 4. Reference image (left), transformed noisy test image (center), and result of their correct mapping (right).

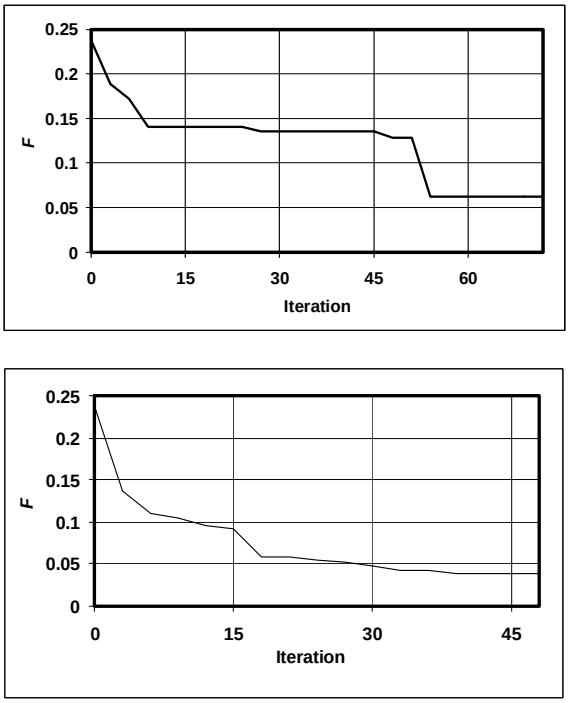

Figure 5. Performance of the classical (top) and the hybrid (bottom) model of evolutionary algorithms.

The purpose of the first group of experiments is to show the effect of the local neighborhood search on the overall performance of evolutionary algorithm. The test set of two 256×256 – pixel synthetic images computed according to formulae given in [1] is shown in Figure 4. The second image in the set is subject to the rigid body transformation defined by the *a priori* known translations *DX* and *DY*, and rotation θ; it has to be mapped onto the first image of the set. Firstly, the classical model of EAs was run using the conventional random genetic operators of crossover, mutation, and selection. Then the classical procedure was augmented with a simple version of a gradient descent local search, to form a hybrid model. During each iteration of the global evolutionary search, local search procedure is applied to every chromosome $\boldsymbol{V}_k$ in the current population in the following manner. A total of six neighbors are drawn at random from the local neighborhood of $\boldsymbol{V}_k$ having a radius of three units along each component of the parameter vector. If a neighbor is found with a better (i.e., smaller) value of the fitness function *F*, it replaces the targeted chromosome $\boldsymbol{V}_k$ in the current population, and algorithm continues the search.

As one can see from the sample results presented in Figure 5, local search significantly improves the overall performance of the algorithm, and helps find a better value of the fitness function $F^* = 0.0581$ in just 18 iterations, as opposed to the value $F^* = 0.0620$ in 54 iterations of the classical EA. The best value found with the hybrid model is $F^* = 0.0387$, at the iteration 39.

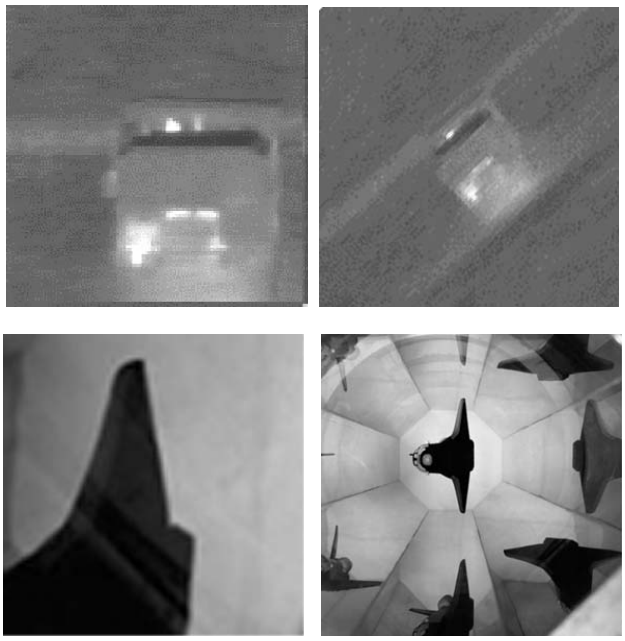

Figure 6. Two sets of test images: a truck set (top row), and a wing set (bottom row). In each set an object (left column) has to be mapped onto a scene (right column).

In the second group of experiments, a study is conducted to compare two customarily used local optimization techniques - the Conjugate gradient method and the Downhill simplex method – on the basis of their robustness and performance. Both methods were implemented as described in [23]. Experimental sets of two 286×286 – pixel images of a truck [28] and two 290×290 – pixel images of an aircraft wing [30], [31] are shown in Figure 6. In both sets, the object is subject to similarity transformation in the 4-dimensional parameter space defined by translations *DX* and *DY*, rotation θ, and isotropic scaling *S*. In

the case of CGM, the first-order forward finite difference scheme is used to compute the gradient *F'* of the fitness function *F*, with respect to each component of the parameter vector ***V***. In five consecutive runs of CGM, the starting point of the search is placed at the distance of 1%, 3%, 6%, 8%, and 9% from the optimum solution, respectively. In the first run of DSM, the five vertices of the initial simplex are evenly placed at the distance of 7% - 11% from the optimum solution. In the second run of DSM, the vertices of the initial simplex are placed around the optimum solution at the equal radius of 10%.

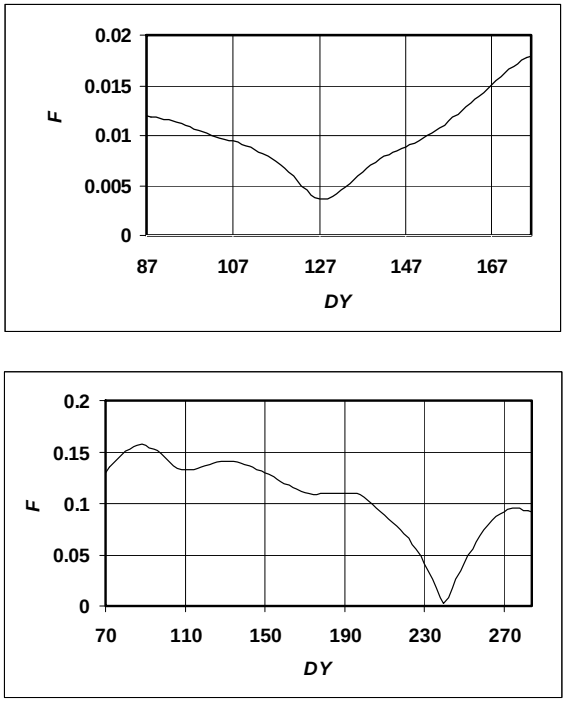

Figure 7. Fitness function *F* along the *y*-axis near the optimum solution for a truck (top), and for a wing (bottom).

Figure 7 shows sample graphs of the fitness function *F* along the *y*-axis in the vicinity of the optimum solution, for both objects. In the case of the truck, the function is convex and sufficiently smooth for evaluating the gradients. The shape of *F* becomes closer to parabolic, as the value of the translation *DY* approaches the optimum point. Based on these geometric properties of the function, one can expect that the derivative-based CGM would exhibit good performance. On the contrary, the fitness function of the wing is significantly non-convex. Only if the point is very near the optimum solution, the shape of *F* becomes close to parabolic. Consequently, one can expect that CGM could perform well only if the starting point of the search would be very close to the optimum.

The results of the computational experiments presented in Tables 1 and 2 support the preliminary arguments. As expected, CGM shows reasonable stability for the truck object, within the tested range of 1% - 9% of the placement of the starting point. The quality of the final solution degrades as the distance increases: the computed minimum fitness value grows from $F^* = 0.00398$ at the 1% distance, to $F^* = 0.00812$ at the 9% distance. In the case of the wing, the performance of CGM quickly deteriorates, as the starting point moves away from the optimum: the computed minimum value of the fitness function increases from $F^* = 0.00599$ at the 1% distance, to $F^* = 0.01604$ at the 3% distance. The algorithm breaks down and quickly diverges beyond the 3% distance range.

	Truck			**Wing**		
	1%	**3%**	**9%**	**1%**	**3%**	**9%**
Fitness ($\times 10^5$)	398	412	812	599	1604	fail
Iterations	58	101	119	60	40	fail
Evaluations:						
Function	91	132	168	78	49	fail
Derivative	61	106	125	63	42	fail
Total	152	238	333	141	91	fail

Table 1. Performance of CGM at the initial distances 1%, 3%, and 9% from the optimum solution.

	Truck		**Wing**	
	7%-11%	**10%**	**7%-11%**	**10%**
Fitness ($\times 10^5$)	391	338	253	308
Iterations	40	137	42	127
Evaluations:				
Function	46	148	50	139
Derivative	-	-	-	-
Total	46	148	50	139

Table 2. Performance of DSM at the initial distance range 7% - 11%, and at the radial distance 10% from the optimum solution.

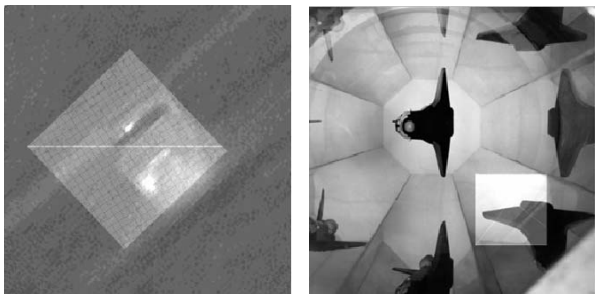

Figure 8. Results of the successful mapping of objects onto original scenes for a truck (left), and for a wing (right).

The Downhill simplex method is not sensitive to the shape of the fitness function, whether convex or non-convex; it displays noticeable robustness in both the truck and the wing cases. The algorithm provides good final results when the initial simplex vertices evenly placed at the 7% - 11% distance, and when the vertices are initially placed around the optimum value at the 10% distance. The performance characteristics of DSM are significantly superior to that of CGM. The computed minimum values for the truck, $F^* = 0.00391$ and $F^* = 0.00338$, at the 7% - 11% and the 10% distance range, respectively, are even better than the value $F^* = 0.00398$ at the 1% distance obtained with CGM. For the wing object, DSM finds very good optimum values $F^* = 0.00253$ and $F^* = 0.00308$ during both runs, whereas CGM fails to obtain any result beyond the 3% initial distance from the optimum. Moreover, the number of function

evaluations with DSM in all considered cases is significantly smaller than the total number of function evaluations obtained with CGM, e.g., 46 at the range 7% - 11%, against 333 at the distance 9%, for the truck object. The correct mapping between the test images found with DSM is shown in Figure 8.

The results of this group of experiments are clearly in favor of using direct search methods, particularly the Downhill simplex method, over gradient-based methods like CGM, as the former can effectively deal with nonlinear, non-convex, numerically computed fitness functions that routinely arise in real world imaging applications.

In the third group of experiments, a modified local search with the proposed two-phase (random / DSM) cyclic procedure is compared with the regular DSM procedure described in [23]. The same test set of a wing shown in Figure 6 is used, with the initial placement of the simplex vertices evenly within the 7% - 11% range from the optimum solution. The results presented in Table 3 show a 32% reduction (from 50 to 34) of the total number of fitness evaluations, when the modified algorithm is used. The optimum value $F^* = 0.00216$ found by the modified version is even slightly better than the value $F^* = 0.00253$ found by the regular DSM. These results indicate that the proposed two-phase modification of the local procedure that includes cyclic random re-evaluation of the simplex vertices, alongside with maintaining the record of the visited points in the simplex neighborhood can speed up the overall convergence of local search while retaining the high quality of the computed optimum values.

	Regular DSM	**Modification**
Fitness ($\times 10^5$)	253	216
Cycles	8	4
Evaluations	50	34

Table 3. Performance characteristics of the regular DSM, and its two-phase cyclic modification.

The purpose of the last group of experiments is to show how utilizing image local response can improve the performance of the local DSM search. The test set includes a 256×256 – pixel image of a scene, and a 236×179 – pixel image of a transformed object [30], [31]. The object is subject to a partial affine transformation A defined by the *a priori* known 5-dimensional parameter vector $\boldsymbol{V} = \{DX, DY, \theta, SX, SY\}$. The transformation A includes the rigid body movement and a local distortion of the object caused by the different values of the scaling factors $SX \neq SY$. Figure 9 presents the images of the scene and the object, as well as the image of the object local response.

A series of 100 runs for the test set is performed, where for each run the vertices of the initial DSM simplex are placed at random in the vicinity of the optimum solution in the following manner. The value of each component of the vector $\boldsymbol{V}$ for each of the six vertices is independently drawn at random, with uniform probability from the (±10%) range of the corresponding domain centered at the component's optimum value. For example, the translation DX for a 256×256 - pixel image has a domain range of 0 – 255. Correspondingly, the value of DX for the image is drawn at random, with uniform probability from the interval (40.0 ± 25.5), i.e., from the interval (14.5, 65.5), where $DX = 40.0$ corresponds to the optimum value of the component DX.

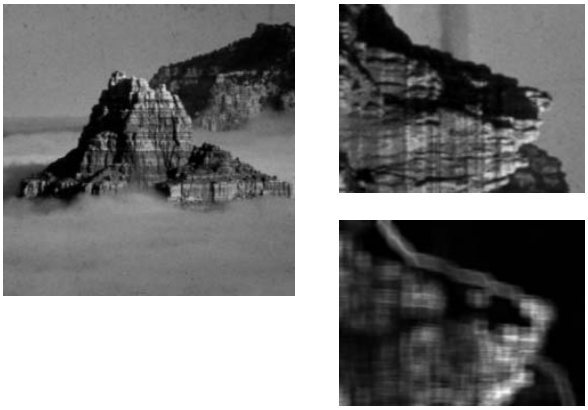

Figure 9. Test image of the original scene (left); the transformed object (top, right); and the object local response (bottom, right).

The series of 100 runs is performed for the regular (i.e., constant) values of the DSM coefficients, and for the modified (i.e., variable) DSM coefficients. The latter are computed according to Equations (5) and (6), using image local response as a means to adaptively control the values of the coefficients. The computed mean values of the search parameters for the test set over all runs are shown in Table 4. The regular algorithm is able to find almost exact parameter values, with the maximum relative error of 1.85%. The optimum values of the search parameters obtained with the modified version of the algorithm are almost identical (within the round-off error) to the values obtained with the regular version. The number of fitness evaluations for the regular and for the modified versions of DSM is presented in Table 5 and Figure 10. As one can see, the fixed values of the DSM coefficients require the most function evaluations, while the use of the response coefficients in the local DSM search results in a significant reduction of over 43% of the number of evaluations. This result clearly indicates that the proposed modified version of the algorithm utilizing the variable coefficients to control the search can improve the algorithm performance, in comparison with the regular version of DSM.

	DX	***DY***	$\boldsymbol{\theta}$	***SX***	***SY***
Regular	40.3	160.3	1.58	2.0	1.5
Modified	40.3	160.3	1.58	2.0	1.5
Exact	40	160	1.571	2.0	1.5

Table 4. Mean values of the search parameters computed with the regular, and with the modified versions of DSM.

	Evaluations	**Reduction rate, %**	**Minimum fitness**
Regular	51	-	0.00235
Modified	29	43.1	0.00257

Table 5. Number of fitness evaluations (mean value), reduction rate, and minimum fitness values computed for the regular, and for the modified versions of DSM.

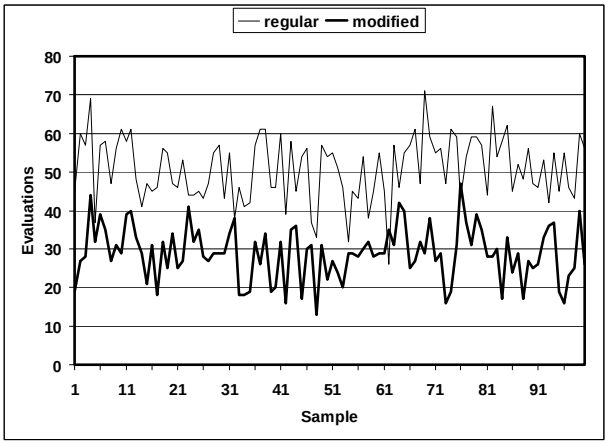

Figure 10. Number of fitness evaluations over 100 runs, for the regular, and for the modified versions of the DSM search.

6. CONCLUSIONS

The paper focuses on efficiency of local search in a Hybrid evolutionary algorithm, with application to optimization problem encountered in electronic imaging. Although the hybrid model can significantly improve the overall performance of evolutionary search, the direct usage of methods of local optimization as simple plug-ins gives rise to a few performance problems. One of the problems is related to a noticeable cost of additional evaluations of fitness function attributed to local search. Another problem stems from the fact that in real world imagery, fitness function commonly has an unknown number of local minima, in which case the computational resources spent on exploring the local neighborhood of a particular chromosome will be wasted if the chromosome happened to belong to one of such local minima. The third problem is associated with the actual distance from the investigated chromosome to the global optimum solution: the distance might not be sufficiently small for the local search to successfully take off and descend to the minimum point.

Computational performance of local search can be potentially improved by applying the following techniques:

- Using direct vs. gradient-based methods, in order to accommodate shape irregularities of fitness function arising in real world imaging applications,
- Adding randomness to a chosen local procedure and periodically re-positioning the search, in order to prevent its convergence to a sub-optimal solution,
- Creating a tree-like structure of local neighborhoods, and engaging memory bookkeeping operations, in order to keep track of the used search space,
- Using cyclic vs. complete local search, in order to cut down on excessive computational cost of local operations, when a chromosome is discarded at a later stage of the global search,
- Incorporating image response analysis, in order to provide the algorithm with a means by which it can autonomously derive problem-specific knowledge and speed up the solution.

A procedure of a two-phase cyclic local search is proposed that incorporates the aforementioned techniques, and improves the computational performance of local search in imaging optimization with HEA. The algorithm maintains a global list of the fittest chromosomes whose neighborhoods are subject to local search. A random search is performed in the neighborhood of each chromosome in the list, with the radius of the neighborhood and the number of the selected neighbors computed as linear functions of time. During the second phase, local search is refined using the Downhill simplex method. To reduce a potential waste of computational resources, DSM performs only a specified number of complete cycles during each call. At regular time intervals, local search switches from DSM back to the random search. Periodic alternating the random and the DSM search continues until either the DSM search successfully terminates, or the chromosome is replaced with a fitter one found by the global evolutionary search.

Engaging image local response helps further reduce the total computational cost of the additional fitness evaluations attributed to the local DSM search. Image local response defines a contraction transformation applied to the vector of the DSM coefficients, in order to adaptively control the shape and the movement of the simplex. The transformation adjusts the length of the vector and, correspondingly, the step size of the simplex movement to the properties of the local fitness landscape.

A series of computational experiments with 2-D grayscale images provide support for the proposed approach. A noticeable reduction of computational cost results from incorporating the techniques described in the paper into the global evolutionary procedure. Moreover, the quality of the final solution does not degrade, in comparison with conventional methods.

7. REFERENCES

[1] R. R. Brooks and S. S. Iyengar, **Multi-sensor fusion: Fundamentals and applications with software**, New York, Prentice Hall, 1998.

[2] G. Dzemyda, V. Saltenis, and A. Zhilinskas, **Stochastic and global optimization**, Kluwer Academic Publishers, 2002.

[3] D. E Goldberg, **Genetic Algorithms in search, optimization, and machine learning**, Addison-Wesley, 1989.

[4] T. Bäck, **Evolutionary algorithms in theory and practice,** New York, Oxford University Press, 1996.

[5] M. Gen and R. Cheng, **Genetic algorithms and engineering optimization**, New York, Wiley, 2000.

[6] D. B. Fogel, **Evolutionary computation: Toward a new philosophy of machine intelligence**, IEEE Press, 2000.

[7] J. M. Fitzpatrick and J. J. Grefenstette, "Genetic algorithms in noisy environments**", Machine Learning**, No. 3, 1988, pp. 101-120.

[8] V. R. Mandava, J. M. Fitzpatrick, and D. R. Pickens, III, "Adaptive search space scaling in digital image registration", **IEEE Transactions on Medical Imaging,** Vol. 8, No. 3, 1989, pp. 251-262.

[9] A. Hill and C. J. Taylor, "Model-based image interpretation using genetic algorithms", **Image and Vision Computing**, Vol. 10, No.5, 1992, pp. 295—300.

[10] B. Turton, T. Arslan, and D. Horrocks, "A hardware architecture for a parallel genetic algorithm for image registration", **Proceedings of the IEE Colloquium on Genetic Algorithms in Image Processing and Vision**, October 1994, pp. 11/1--11/6.

[11] C. A. Ankenbrandt, B. P. Buckles, and F. E Petry, "Scene recognition using genetic algorithms with semantic nets", **Pattern Recognition Letters**, Vol. 11, No. 4, 1990, pp. 231-304.
[12] B. P. Buckles and F. E Petry, "Cloud identification using genetic algorithms and massively parallel computation", **Final Report**, Grant No. NAG 5-2216, Center for Intelligent and Knowledge-based Systems, Dept. of Computer Science, Tulane Univ., New Orleans, LA, June 18, 1996.
[13] D. Prabhu, B. P. Buckles, and F. E Petry, "Genetic algorithms for scene interpretation from prototypical semantic description", **International Journal of Intelligent Systems**, Vol. 15, No. 10, 2000, pp. 901-918.
[14] W. Siedlecki and J. Sklansky, "A note on genetic algorithms for large-scale feature selection", **Pattern Recognition Letters**, Vol. 10, No. 5, 1989, pp. 335-347.
[15] R. Leardi, R. Boggia, and M. Terrile, "Genetic algorithms as a strategy for feature selection", **Journal of Chemometrics**, Vol. 6, No. 5, 1992, pp. 267-281.
[16] M. Kudo and J. Skalansky, "Comparison of algorithms that select features for pattern classifiers", **Pattern Recognition**, Vol. 33, No. 1, 2000, pp. 25-41.
[17] W. E. Hart and R. K. Belew, "Optimization with genetic algorithm hybrids that use local search", **Adaptive individuals in evolving populations: models and algorithms: Proceedings of Santa Fe Institute studies in the sciences of complexity**, R. K. Belew and M. Mitchel, eds., Vol. 26, 1996, pp. 483-496, Reading, MA: Addison-Wesley, 1996.
[18] J. A. Joines and M. G. Kay, "Utilizing hybrid genetic algorithms", **Evolutionary optimization**, R. Sarker, M. Mohammadian, and X. Yan, eds., pp. 199-228, Boston, MA: Kluwer Academic Publishers, 2002.
[19] J. H. Holland, **Adaptation in natural and artificial systems.**, University of Michigan Press: Ann Arbor, 1975; MIT Press (2nd edition), 1992.
[20] M. Mitchel, **An introduction to genetic algorithms**, MIT Press, 1996.
[21] R. L. Haupt and S. E. Haupt, **Practical genetic algorithms**, John Wiley & Sons, 1998.
[22] **Evolutionary computation 1: Basic algorithms and operators**, T. Bäck, D.B. Fogel, and T. Michalewicz, eds., Bristol, Philadelphia: Institute of Physics Publishing, 2000.
[23] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, **Numerical Recipes in C**, Cambridge, UK: Cambridge University Press, 1988.
[24] T. G. Kolda, R. M. Lewis, and V. Torczon, "Optimization by direct search: New perspectives on some classical and modern methods**", SIAM Review**, Vol. 45, No. 3, 2003, pp. 385-482.
[25] J. A. Nelder and R. Mead, "A simplex method for function minimization**", Comput. J.**, No. 7, 1965, pp. 308-313.
[26] M. H. Wright, "Direct search methods: Once scorned, now respectable", **Numerical Analysis 1995: Proceedings of the 1995 Dundee Biennial Conference in Numerical Analysis**, D. F. Griffiths and G. A. Watson, eds., pp. 191-208, Addison Wesley Longman, Harlow, UK, 1996.
[27] F. Glover and M. Laguna, **Tabu search**, Kluwer Academic Publishers, 1997.
[28] I. Gertner and I. V. Maslov, "Using local correction and mutation with memory to improve convergence of evolutionary algorithm in image registration", **Automatic Target Recognition XII: Proceedings of SPIE**, Firooz A.Sadjadi, ed., Vol. 4726, pp. 241-252, SPIE, 2002.
[29] I. V. Maslov, "Improving local search with neural network in image registration with the hybrid evolutionary algorithm", **Intelligent computing: Theory and applications: Proceedings of SPIE**, K. L. Priddy and P. J. Angeline, eds., Vol. 5103, pp. 166-177, SPIE, 2003.
[30] http://lisar.larc.nasa.gov (public domain).
[31] http://www.photolib.noaa.gov (public domain).