

MATCHING OF OBJECTS NODAL POINTS IMPROVEMENT USING OPTIMIZATION

Luísa F. Bastos

*LOME - Laboratório de Óptica e
Mecânica Experimental
INEGI - Instituto de Engenharia
Mecânica e Gestão Industrial
Porto, Portugal
lbastos@fe.up.pt*

João M. R. S. Tavares

*LOME, INEGI, DEMEGI - Departamento de
Engenharia Mecânica e Gestão Industrial
FEUP - Faculdade de Engenharia da
Universidade do Porto
Porto, Portugal
tavares@fe.up.pt*

ABSTRACT

The main objective of this work was to improve a previously developed object matching methodology. This overall methodology includes: a modeling phase; followed by a modal analysis; the construction of a matrix that relates both sets of objects points; and the matching phase.

The previously implemented matching phase is based on a local search; with this solution the relation between objects nodes (points) are not considered. To overcome this, we implemented a new matching solution, using optimization techniques, based on a global search. This solution is compared with the previous one allowing the verification of the results improvement.

The local and the global search methods used in this work allow only matches of the usual type “one to one”. However, there are situations when this type of matches is not the most adequate, since it can imply loss of information. To avoid this problem, we developed a new algorithm applicable to contour objects that finds satisfactory matches of type “one to many” or vice-versa.

INTRODUCTION

Establishing correspondences between objects is an essential step of some computational vision domains, as for example, analyzing image objects movement/deformation.

The resolution methods for the matching problem usually include restrictions that avoid incorrect matches according to the considered criteria. Examples of those restrictions are [1]: the order [2, 3], epipolar restrictions (rigidity restrictions) [2, 3], uniqueness [4], visibility [5], and proximity. To determine the matches it can be

used, for example, image correlation (i.e. images similarity is assumed) [1, 6, 7], the points’ proximity [7, 8], or the disparity fields smoothness [2].

The matching problem can also be interpreted as an optimization problem, where the objective function can depend, for example, of any image relationship mentioned in the last paragraph, and the restrictions have to form a non-empty space of possible solutions for the optimization problem. This solution can use: dynamic programming [2], graph search [3], bipartite graph matching [9], convex minimization [7], etc. Non-optimal approaches include: greedy algorithms [10], simulated annealing [11], relaxation [4], etc.

In this work, the matching process is based on the construction of a matrix (the affinity matrix) that is obtained by one of the two different methodologies previously implemented [12, 13]: (1) using the shape modal analysis [14]; or (2) using the finite elements method and modal analysis [15]. Both methods establish the correspondences from the displacement analysis of each point in the respective modal space.

These methodologies can be used to satisfactorily match 2D or 3D objects. However, such as they were initially proposed, the matching phase has two considerable disadvantages: 1) the matches are established using a local approach, i.e. the relationship between the points of a same object is not considered; 2) when the considered objects have a different number of elements, the excess points are not matched, since in this approach the contemplated correspondence type is the usual “one to one”.

Here, the first disadvantage is overtaken [16] using global optimization methods in the matching process, after adjusting this problem to

a classical assignment problem [17]. This new methodology improved the results in several aspects (number of matches, robustness, etc.).

The problem of the excess points is solved with the adaptation of matches of type “one to many” and vice versa, using neighborhood criterions applicable to contour objects. The developed algorithm matches successfully the excess points in the examples considered.

BASE METHODOLOGIES

In this section are presented the two adopted modeling methods (geometrical and physical) that can be used in the construction of the affinity matrix. After the model derivation, the construction of the affinity matrix and the local approach, previously used for the determination of the matches, is explained.

Model Construction

Geometrical Methodology. This method [12, 13, 14] analyzes individually each object for its shape modes extraction and uses them to establish the correspondences. Basically, the modes codify the object shape based on the distances between its points.

First, let's consider the problem of determining the shape modes of an object with m points of coordinates X_i , with $i = 1, 2, \dots, m$. A square proximity matrix $[H]$ is built with a Gaussian weighted measure:

$$H_{ij} = e^{-(r_{ij})^2/(2\sigma^2)} \quad (1)$$

where $(r_{ij})^2 = \|X_i - X_j\|^2$ is the Euclidian distance between points i and j of the mentioned object.

After the construction of the proximity matrix, the eigenvalues and eigenvectors of $[H]$ are determined solving the generalized system:

$$[H] = [\Phi][\Omega][\Phi]^T \quad (2)$$

where $[\Omega]$ is the ordered eigenvalues diagonal matrix and $[\Phi]$ is the eigenvectors matrix ordered by the associated eigenvalues in columns. Each line vector of matrix $[\Phi]$ is called a characteristic vector.

This computational process is executed for the objects to match, t and $t+1$. The greatest advantage of this method is the computational

efficiency, providing good results for several applications within a low execution time [13].

Figure 1 schematizes the overall methodology using the geometrical approach.

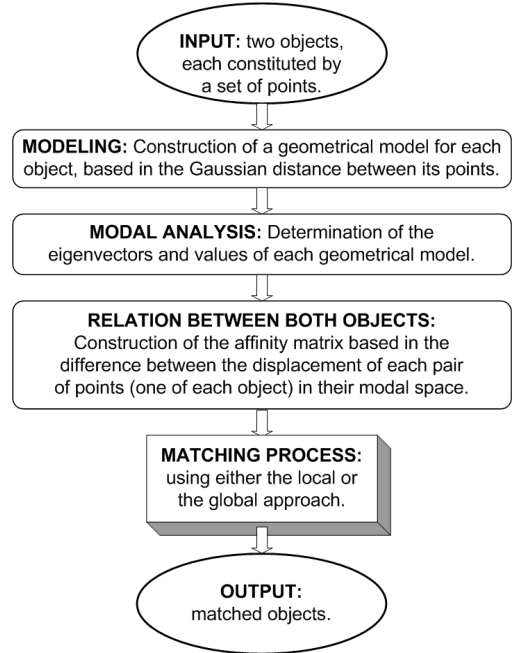


Figure 1. Diagram of the overall methodology using the geometrical approach.

Physical Methodology. This approach [12, 15] uses the finite elements method to construct a physical model for each object. In this work we used the Sclaroff's isoparametric finite element [12, 13, 15]. The greatest advantage of this method is the attribution of a physical behaviour to the object to be modelled, through the consideration of a virtual material [13].

The dynamic equilibrium equation system for the finite elements model can be written as:

$$[M]\{\ddot{U}\} + [C]\{\dot{U}\} + [K]\{U\} = \{R\}, \quad (3)$$

where $\{U\}$ is the displacement components vector; $\{\dot{U}\}$ and $\{\ddot{U}\}$ are the first and second time derivatives of $\{U\}$; $[M]$, $[C]$ and $[K]$ are the element mass, damping and stiffness matrices, respectively; and $\{R\}$ is the load vector. This system of equations can be decoupled by posing the equations in a basis defined by the M-orthonormalized eigenvectors of $[M]^{-1}[K]$, [12, 15]. These eigenvectors and values are computed

solving the following system:

$$[K][\Phi] = [M][\Phi][\Omega], \quad (4)$$

where $[\Phi]$ is the eigenvectors matrix sorted by the associated frequencies in columns; and $[\Omega]$ is the sorted eigenvalues diagonal matrix.

For an object with m nodes, the column vector $\{\phi\}_{i,j}$ of $[\Phi]$ describes the modal displacement of each model node, due to the vibration mode j , while the row vectors $\{\phi\}_{i,\cdot}$ and $\{\phi\}_{i+m,\cdot}$ (and $\{\phi\}_{i+2m,\cdot}$ for 3D objects) are the characteristic vectors and together describe the location of node i in the modal space.

Figure 2 schematizes the overall methodology using the physical approach.

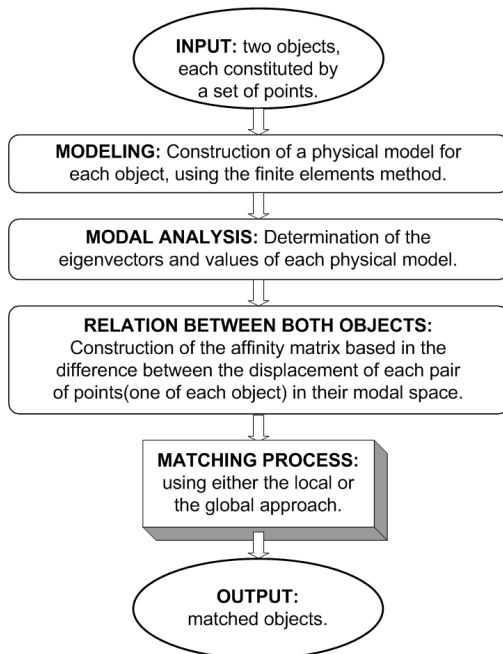


Figure 2. Diagram of the overall methodology using the physical approach.

Affinity Matrix Construction

After the construction of a model for each object, its eigenvectors are determined and stored in matrix $[\Phi]$ (in columns), [12, 13, 14, 15].

The final step to construct the affinity matrix, $[Z]$, consists in the computation of the Euclidian distances between the characteristic vectors (line vectors of $[\Phi]$) of both objects:

$$Z_{ij} = \sum_{k=0}^{dim} \|\{\phi_t\}_{i+k \cdot m,\cdot} - \{\phi_{t+1}\}_{j+k \cdot n,\cdot}\|^2 \quad (5)$$

where $dim = 1$ if the objects are bi-dimensional or $dim = 2$ if the objects are tri-dimensional, and m and n are the number of nodes of objects t and $t+1$, respectively. In $[Z]$, Z_{ij} quantifies the confidence of the match between points i of object t and j of the object $t+1$.

After the construction of this affinity matrix, the matches are determined by the similarity of the displacements of each par of points (one of each object) in their respective modal space. (As lower is an element (i, j) of matrix $[Z]$, higher is the probability of the match between points i and j (of objects t and $t+1$, respectively) being adequate.)

Previous matching approach (Local)

The previously implemented solution for the determination of correspondences is based on a local approach [12, 13, 14, 15] and basically it consists in searching sequentially, in the affinity matrix, the lowest element in each line and matching it with the associated column, if it is also the lowest element in that column and if it is below a pre-established threshold. This last restriction prevents matches between points that have a weak affinity, avoiding two objects with significant differences (of shape, for example) to have a good matching result. As mentioned, the major disadvantage of this method lays in the fact that each point is processed individually, disregarding the global model. To overcome this disadvantage, a new approach for the matching phase was developed, using global optimization techniques. This new approach showed to improve the matching results and it's presented in the next section.

MATCHING WITH OPTIMIZATION

The matching problem, considering only matches of type "one to one", can be written as a classical assignment problem, witch is a particular case of an optimization problem [16, 18]. In the formulation of this problem it's assumed that: the number of points of both objects is the same – n ; it is known the assignment cost of each pair of points $(i, j) - Z_{ij}$, where i is a point of object t and j is a point of object $t+1$. As the notation tries to evidence, this assignment cost is equal to the element of matrix $[Z]$, in line i and column j .

The assignment problem initially appeared to

mathematically formulate the need to distribute n jobs/works by n tasks, where: each job/work had to be assigned to one and only one task and vice-versa; the assignment total cost had to be minimized. Here, the jobs/works are the points of object t and the tasks are the points of object $t+1$. So, for the mathematical formulation of this problem, let's consider:

$$x_{ij} = \begin{cases} 1 & \text{if point } i \text{ of } t \text{ is assigned} \\ & \text{to point } j \text{ of } t+1 \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

with $i, j = 1, 2, \dots, n$. Next expressions follow the structure of a mathematical programming problem, with an objective function and a set of restrictions in the problem's variables:

$$\text{minimize } f = \sum_{i=1}^n \sum_{j=1}^n Z_{ij} x_{ij} \quad (7)$$

$$\text{subject to } \sum_{j=1}^n x_{ij} = 1, \text{ with } i = 1, 2, \dots, n \quad (8)$$

$$\sum_{i=1}^n x_{ij} = 1, \text{ with } j = 1, 2, \dots, n \quad (9)$$

$$\text{and } x_{ij} \in \{0, 1\}, \forall i, j. \quad (10)$$

In equation (7), function f takes the value of the assignment total cost. The first restriction (8) forces each point in object t to be assigned to one and only one point in object $t+1$. The second restriction (9) forces each point in object $t+1$ to be assigned to one and only one point in object t . The third and last restriction (10) forces the problem's variables, x_{ij} ($i, j = 1, 2, \dots, n$), to take one of the two possible values $\{0, 1\}$.

To solve the assignment (matching) problem we used three algorithms: the *Hungarian* method [17, 19]; the *Simplex* method for flow problems [19, 20]; and the *LAPm* algorithm [19, 21]. The *Hungarian* method is the most well known method for the resolution of the assignment problem. The *Simplex* method for flow problems solves problems with less rigid restrictions than the assignment problem, but where this last can be included as a special case. The *LAPm* method is a considerably recent algorithm developed to solve classical assignment problems.

Once the optimal solution is found, the matches that exceed a pre-established threshold are eliminated from that solution, as in the local matching approach. The relevance of this

restriction is higher with this methodology, because a solution of the assignment problem always has the maximum number of matches of type "one to one", due to the problem restrictions.

EXCESS POINTS MATCHING

Case the number of points of the objects to be matched is different, with the usual matching restriction of type "one to one" there will be points that won't be matched. Moreover, the problem data will not verify the presumption of the classical assignment problem (the number of points has to be the same for both objects).

In the first step of the adopted solution, fictitious points are added to the model with fewer elements. This way, the matching problem is written as a classical assignment problem, verifying its presumption. (The matching costs associated to fictitious points are all set to zero.)

The real points matched with the fictitious points, after the optimization phase, are called excess points. These excess points are subsequently matched with real points, using a new implemented algorithm (*ADCom*). As a result, matches of type "one to many" or vice-versa are allowed.

This algorithm is based on neighbourhood criterions and can be succinctly described as follows: Each one of the excess nodes is fitted between the nearest matched points (of the same object); From the matches of these neighbours into the other object, it's determined the best correspondence, minimizing the cost, considering that neighbours should continue neighbours, and shouldn't be allowed cross correspondences. As in the optimization phase, the obtained matches are only considered if they respect the pre-established threshold.

RESULTS

In this section are presented some experimental results obtained with the described methods.

The results analysis present a comparison between the previously developed methodology for the determination of correspondences, based on a local approach (local search), and the new proposed methodology, based on a global approach using optimization techniques (global search). The integrated optimization algorithms are not analyzed individually, because the number of matches and the total matching cost were the same for all of them in all the presented experiences. However, the algorithms efficiency

is commented in the conclusions.

The examples proceeding from synthetic images allow the quality verification of the obtained results through the confirmation of the points order. With contour objects that verification is visually facilitated. For each example, besides the matching images result, are also presented some numerical results including: the matching algorithms used (Matching Algo.); the number of matches obtained (Nr Match.); the total matching cost, i.e. the value of f in equation (7), divided by the total number of matches, resulting in the mean cost per match (Cost/Mat); and the reference to the figures that illustrate each result (Fig.).

The first 2D example is constituted by two synthetic sets of points of a “clover” contour (each composed by 65 points) with a rigid transformation between them. Figure 3 presents the matching results using the geometrical modeling methodology and, the local matching solution (Figure 3(a)) and the global matching solution (Figure 3(b)). Table 1 presents some numerical results of this example.

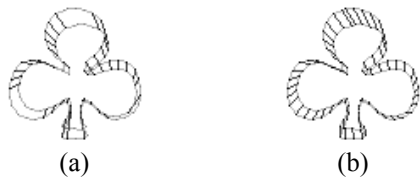


Figure 3. Matched objects (“cloves”), using the method (a) local search and (b) global search.

Table 1. Numeric matching results of the first 2D example.

Matching Algo.	Nr Match.	Cost/Mat	Fig.
local search	40 (62%)	$5.8775 \cdot 10^{-3}$	3 (a)
global search	65 (100%)	$8.4392 \cdot 10^{-3}$	3 (b)

In Figure 3 are visible several areas in the matched “cloves” where satisfactory correspondences were found in 3(b) - global search, but not in 3(a) - local search. The results from Table 1 show that with the previous matching method (3(a)) were found 62% of the possible matches. With the new approach (3(b)), based on optimization techniques, all the matches were found and 100% of them were correct.

It was also observed that, if the rigid

transformation between two objects is too high, it's difficult to match them satisfactorily; this is a problem associated with the adopted modeling methodology [13].

Next 2D example is composed by two contours (with 99 and 98 points) from two different real retinal images. Figures 4 and 5 show the matching results using the geometrical modeling methodology and, the local solution (Figure 4) and the global solution (Figure 5). Table 2 presents some numerical results of this example.



Figure 4. Matched objects (“retina”), using the local search method.



Figure 5. Matched objects (“retina”), using the global search method.

Table 2. Numeric matching results of the second 2D example.

Matching Algo.	Nr Match.	Cost/Mat	Fig.
local search	67 (68%)	$9.2296 \cdot 10^{-6}$	4
global search	97 (99%)	$3.0201 \cdot 10^{-5}$	5

In Figures 4 and 5 is perceptible that the global search algorithm (Figure 5) found a higher

number of matches than the local search algorithm (Figure 4). It is also visible that in Figure 4 there are incorrect matches (see arrows) that are satisfactory in Figure 5.

In this example, weren't found all the possible matches of type "one to one" even when the global search algorithm was used (see Table 2). This is due to the pre-established threshold. If this parameter wasn't applied, the matching result would be the one showed in Figure 6 (the missing match is the one marked and is incorrect).



Figure 6. Matched objects ("retina"), using the global search method, without the matching threshold parameter applied.

Next 2D example intends only to show one application of the *ADCom* algorithm. In Figure 7 is presented the matching result between two contours (with 32 and 28 points) extracted from two real heart images, using the physical modeling methodology, rubber as the virtual material, and the global search and *ADCom* matching algorithms.

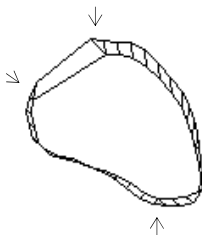


Figure 7. Matched objects ("hearts"), using the global search method and the *ADCom* algorithm.

The arrows in Figure 7 point out some visible

matches of type "many to one" established with the *ADCom* algorithm. All points of both objects were satisfactorily matched.

The last example is composed by two 3D surfaces (with 117 and 112 points) obtained from two real images of a dynamic pedobarography exam [22]. Some numerical results obtained in the matching process are presented in Table 3. In Figures 8 to 11 are presented the matching results, using either the local search or the global search algorithms. For this example was used the physical modeling methodology and rubber as the virtual material.

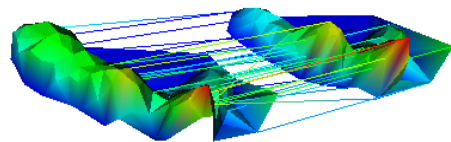


Figure 8. Matched objects ("pedobarography"), using the local search method (front view).

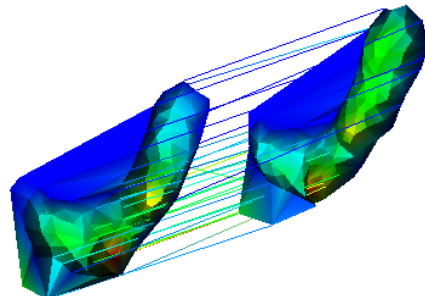


Figure 9. Matched objects ("pedobarography"), using the local search method (back view).

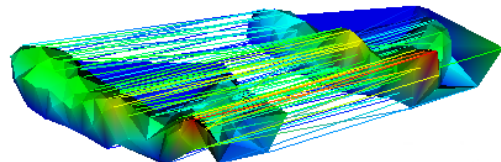


Figure 10. Matched objects ("pedobarography"), using the method global search (front view).

In this example, not all the matches found are satisfactory, independently of the matching methodology applied. This indicates the need for more restrictions to prevent wrong matches. However, the global search algorithm determines 100% of the matches, being most of them satisfactory, while using the local search

algorithm only 29% of the matches are found and, as mentioned, not all satisfactory.

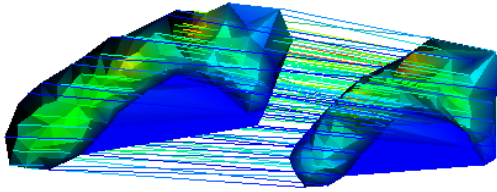


Figure 11. Matched objects (“pedobarography”), using the method global search (back view).

Table 3. Numeric matching results of the 3D example.

Matching Algo.	Nr Match.	Cost/Mat	Fig.
local search	32 (29%)	0.1684	8, 9
global search	112 (100%)	0.1962	10, 11

For all presented examples, the cost per match is slightly higher for the global search than for the local search matching algorithm. However, the final result shows to be more satisfactory for the global search relatively to matching number (up to 70%) and also to correspondences quality.

CONCLUSIONS

The several experimental tests carried through, some reported in this paper, allow the presentation of some observations and conclusions.

The proposed two object matching search methodology, based on optimization techniques, (when compared with the one previously developed, based on a local search) obtained always an equal or higher number of satisfactory matches. It was also observed that the number of matches is independent from the optimization method that we have considered.

In some experimental results, when the local matching search method was applied, the construction of the affinity matrix, using one of the modeling methodologies presented, had to be careful, in order to obtain a higher number of satisfactory matches. In those same examples, the use of an optimization method, beyond the good matching results, revealed less sensible to the referred affinity matrix construction. This suggests that, with the proposed search approach, the overall methodology can become easier to handle and more adjustable to different

applications.

In some practical experiences, the number of eigenvectors used to obtain a satisfactory matching result was inferior using optimization methods. This suggests that using the proposed matching search method the total computational effort diminishes for the overall methodology.

Although not presented in this paper, the three optimization algorithms considered (*Hungarian*, *Simplex* for flow problems and *LAPm*) were compared in several experimental results. In each of the considered examples, the number of matches and the total matching cost were always the same; yet, the execution time was different. Regarding this last outcome, the implemented matching search algorithm that uses the *Hungarian* method was the slowest. This is due to the way the implementation used finds the solution for each iteration (a new solution of n matches is determined, while the other methods change only some matches in each iteration). Contrarily, the *Simplex* algorithm for flow problems revealed the most efficient. The execution time of *LAPm* algorithm was higher than the one for *Simplex* algorithm for flow problems, even being a specific algorithm for this type of problems. This can be dew to the interval in which lay the elements of the affinity matrix, $[0; 1]$, since when this algorithm was tested in [23] it revealed the most efficient when the considered interval was $[1; 100]$.

In the several performed experimental tests, the implemented algorithm for the excess points matching, when applied to contour objects, finds satisfactory matches.

PERSPECTIVES OF FUTURE WORK

The use of optimization techniques in object matching seems to be a promising research area in computational vision.

One possible extension of this work is the application of optimization techniques to objects matching along image sequences. This would also allow the integration in the matching process of the information that could be obtained along the sequence.

The description of the matching problem as an assignment problem leads to a simple formulation. However, there are restrictions that can not be incorporated in this problem. Examples of these restrictions are the neighborhood criterions that are considered in *ADCom* algorithm. The incorporation of these restrictions in the optimization of the matching problem could

be another possible extension of this work.

It could also be considered in future work the hypothesis of allowing matches of type "many to one" and vice-versa in the optimization problem.

ACKNOWLEDGEMENTS

Our thanks to Professor A. Volgenant for making available the source code of his optimization algorithm (*LAPm*) used in this work.

REFERENCES

1. J. L. Maciel and J. P. Costeira, *A global solution to sparse correspondence problems*, IEEE Transactions on Pattern Analysis and Machine Intelligence **25** (2003), no. 2, 187-199.
2. Y. Ohta and T. Kanade, *Stereo by intra- and inter-scanline search using dynamic programming*, IEEE Transactions on Pattern Analysis and Machine Intelligence **7** (1985), no. 2, pp. 139-155.
3. S. Roy and I. J. Cox, *A maximum-flow formulation of the n-camera stereo correspondence problem*, International Conference on Computer Vision (ICCV'98), 1998, pp. 492-499.
4. S. Gold, A. Rangarajan, C. P. La, S. Pappu and E. Mjolsness, *New algorithms for 2D and 3D point matching: Pose estimation and correspondence*, Pattern Recognition **31** (1998), no. 8, pp. 1019-1031.
5. C. Silva, "3D motion and dense structure estimation: Representation for visual perception and the interpretation of occlusions," *Instituto Superior Técnico*, Universidade Técnica de Lisboa, 2001.
6. P. H. S. Torr, "Outlier detection and motion segmentation," *Engineering Department*, University of Oxford, 1995.
7. B. D. Lucas and T. Kanade, *An iterative image registration technique with an application to stereo vision*, 7th International Joint Conference in Artificial Intelligence (IJCAI '81), 1981, pp. 674-679.
8. Z. Zhang, "Iterative point matching for registration of free-form curves," INRIA, 1992, p. 42.
9. G. Fielding and M. Kam, *Weighted matchings for dense stereo correspondence*, Pattern Recognition **33** (2000), no. 9, pp. 1511-1524.
10. M. S. Wu and J. J. Leou, *A bipartite matching approach to feature correspondence in stereo vision*, Pattern Recognition Letters **16** (1995), pp. 23-31.
11. J. P. P. Starink and E. Backer, *Finding point correspondences using simulated annealing*, Pattern Recognition **28** (1995), no. 2, pp. 231-240.
12. J. M. R. S. Tavares, J. Barbosa and A. Padilha, *Determinação da correspondência entre objectos utilizando modelação física*, 9º Encontro Português de Computação Gráfica, Marinha Grande, 2000.
13. J. M. R. S. Tavares, "Análise de Movimento de Corpos Deformáveis Usando Visão Computacional," *Faculdade de Engenharia*, Universidade do Porto, 2000.
14. L. S. Shapiro and J. M. Brady, *Feature-based correspondence: An eigenvector approach*, Image and Vision Computing **10** (1992), no. 5, pp. 283-288.
15. Stan Sclaroff and A. Pentland, *Modal matching for correspondence and recognition*, IEEE Transactions on Pattern Analysis and Machine Intelligence **17** (1995).
16. L. F. Bastos and J. M. R. S. Tavares, *Optimization in modal matching for correspondence of objects nodal points*, 7th Portuguese Conference on Biomedical Engineering (BioEng'2003), Lisbon, 2003.
17. F. S. Hillier and G. J. Lieberman, *Introduction to operations research*, McGraw-Hill International Editions, 1995.
18. C. H. Papadimitriou and K. Steiglitz, *Combinatorial optimization - algorithms and complexity*, Prentice-Hall, Inc., 1939.
19. L. F. Bastos, "Optimização da Determinação das Correspondências entre Objectos Deformáveis no Espaço Modal," *Faculdades de Engenharia e Ciências*, Universidade do Porto, 2003.
20. A. Löbel, "Mfc - a network simplex implementation," Konrad-Zuse-Zentrum für Informationstechnik Berlin, Division Scientific Computing, Department Optimization, 2000.
21. A. Volgenant, *Linear and semi-assignment problems: A core oriented approach*, Computers and Operations Research **23** (1996), no. 10, pp. 917-932.
22. J. M. R. S. Tavares, *Matching Image Objects in Dynamic Pedobarography*, 11th Portuguese Conference on Pattern Recognition (RecPad 2000), Porto, 2000.
23. M. dell'Amico and P. Toth, *Algorithms and codes for dense assignment problems: The state of the art*, Discrete Applied Mathematics **100** (2000), no. 1-2, pp. 17-48.