Available online at www.sciencedirect.com
ScienceDirect
Image and Vision Computing 25 (2007) 1824–1832
www.elsevier.com/locate/imavis

Automatic selection of MRF control parameters by reactive tabu search

U. Castellani, A. Fusiello, R. Gherardi *, V. Murino

Dipartimento di Informatica, Università di Verona, Strada Le Grazie, 15, 37134 Verona, Italy

Received 22 February 2006; received in revised form 22 August 2006; accepted 19 April 2007

Abstract

This paper presents an optimization technique to automatically select a set of control parameters for a Markov random field. The method is based on the reactive tabu search strategy, and requires to define a suitable fitness function that measures the performance of the MRF algorithm with a given parameters set. The technique is applied to stereo matching thanks to the availability of ground truth disparity maps. Experiments with synthetic and real images illustrate the approach.

Keywords: Parameter estimation; Markov random fields; Reactive tabu search

1. Introduction

Markov random fields models have been successfully used in the recent past [1,2] for a wide variety of early vision problems such as image restoration, motion estimation, stereo matching and segmentation. These methods are usually structured around a functional that balances observations consistency with one or more regularization terms, thus leaving at least one free parameter to be chosen. Most of the current MRF algorithms requires a set of hard-coded or user-specified parameters, whose manual search using trial and error is often time consuming and non-repeatable.

In this paper we propose a technique capable of automatic selection of the "best" parameters, based on an optimization algorithm and a suitable fitness function that measures the performance of the MRF algorithm with a given parameters set.

We apply our method to R-SMW (Relaxed Symmetric Multiple Windows), a probabilistic stereo matching algorithm [3], where the winner-takes-all approach of the Symmetric Multiple Window (SMW) algorithm [4] is relaxed by exploiting the non-determinism of the MRF.

The functional has two free parameters that in this paper are computed as the result of an optimization based on the reactive tabu search [5,6], which mitigates the problem of local minima trapping while driving the search to unexplored regions of the solution space. The fitness function is defined by comparing the output disparity with a ground truth.

Even if the resulting stereo algorithm do not outperform recent algorithms based on graph-cut (see [7,8] for a review), the methodology for the estimation of the free parameters of a MRF functional is interesting in itself. Notably, we demonstrate that images with similar characteristics share similar parameters, enabling us to calculate nearly optimal parameters for whole classes of pictures, even when ground truth is not available.

Similar approaches, based on Genetic Algorithms, have been proposed in the past, focusing on different applications [9,10].

In the following, an overview of the R-SMW method proposed in [3] is briefly described (Section 2). Then, in Section 3, an introduction on the tabu search paradigm is explained. The proposed method is detailed in Section 4, where we clarify how the tabu search is applied to the parameter optimization problem outlined. Experimental results on synthetic and real images are reported in Section 5. Finally, conclusions are drawn in Section 6.

* Corresponding author. Tel.: +39 045 8027049; fax: +39 045 8027068.
E-mail address: gherardi@sci.univr.it (R. Gherardi).

0262-8856/$ - see front matter © 2007 Published by Elsevier B.V.
doi:10.1016/j.imavis.2007.04.002

2. The R-SMW algorithm

Before describing the R-SMW algorithm, we shall review the SMW algorithm, upon which the former is based, and the principles of Markov random fields, in order to make the paper self-consistent.

2.1. The stereo process

The matching process consists in finding the element (a point, region, or generic feature) in the right image which is most similar, according to a similarity metric, to a given element in the left image.

In the simple *block-matching* method, similarity scores are computed for every pixel in the left image, by comparing a fixed window centered on the current pixel with a window in the right image, shifted along the raster scan line. It is customary to use the Euclidean distance, or Sum of Squared Differences (SSD), as a (dis)similarity measure. The computed disparity is the one that minimizes the SSD error.

Several factors make the correspondence problem difficult. A major source of errors in computational stereo are occlusions, i.e., points not belonging to any conjugate pairs, although they help the human visual system in detecting object boundaries. There are two key observations to cope with the occlusions issue. The first is that matching is not a symmetric process. Actually, when searching for corresponding elements, only the visible points in the reference image (usually, the left image) are matched. The second observation relies on the fact that in many real cases a disparity discontinuity in one image corresponds to an occlusion in the other image. Some authors [11,12] use the former consideration to validate the matching (left–right consistency). In other cases [13,14], the latter is used to constrain the search space.

Even under simplified conditions, it appears that the choice of the window size is critical. A too small window is noise-sensitive, whereas an exceedingly large one acts as a low-pass filter, and is likely to miss depth discontinuities. Consider the case of a piecewise-constant surface: points within a window covering a surface discontinuity come from two different planes, therefore a single "average" disparity cannot be assigned to the whole window without making a manifest error.

This problem is addressed effectively – although not efficiently – by the Adaptive Window algorithm [15], and by the simplified version of the multiple window approach, introduced in [16,13].

In [4], a new algorithm has been proposed that computes disparity by exploiting both the multiple window approach and the left–right consistency constraint. For each pixel, SSD matching is performed with nine 7×7 windows with different centers: the disparity with the smallest SSD error value is retained. The idea is that a window yielding a smaller SSD error is more likely to cover a constant depth region.

The multiple windows approach can be regarded as a robust technique able to fit a constant disparity model to data consisting of piecewise-constant surface, i.e., capable of discriminating between two different populations. Occlusions are also detected, by checking the left–right consistency and suppressing in-feasible matches accordingly.

The R-SMW algorithm [3] introduces a relaxation of the SMW algorithm using MRF. Both the multiple windows and the left–right consistency features are kept but all these requests are cast in terms of cost functions to be minimized over a disparity field.

2.2. Markov random fields

A MRF is defined on a finite lattice field I of elements i called sites. Let us define a family of random variables $D = \{D_i = d_i,\ i \in I\}$, and let us suppose that each variable may assume values taken from a discrete and finite set.

In general, when an MRF model is applied to computer vision problems, the pixels are the sites, d_i is the grey level of the pixel i, and the image is interpreted as the realization of the discrete stochastic process D_i.

The Markov property states that the conditional probability $P\ (d_i|d_{I-\{i\}})$ depends only on the values on the neighbouring set of i, N_i (see [17]).

Thanks to the Hammersley–Clifford theorem, that establishes the Markov–Gibbs equivalence between MRFs and Gibbs random fields [18], the probability distribution of a *realization* of the field takes the following form:

$$P(d) = Z^{-1} \cdot \mathrm{e}^{-\beta \cdot U(d)} \tag{1}$$

where Z is a normalization factor called partition function, β is a parameter called temperature and $U(d)$ is the energy function, which can be written as a sum of local energy potentials V_c dependent only on the cliques c (local configurations) associated with the neighbouring set for every field site (whose dependence is omitted in the formula for simplicity):

$$U(d) = \sum_{c \in C} V_c(d) \tag{2}$$

In general, given the observation g, the posterior probability $P(d|g)$ can be derived from the Bayes rule by using the a-priori probability $P(d)$ and the conditional probability $P(g|d)$. The problem is solved computing the estimate d according to a Maximum A-Posteriori (MAP) probability criterion. Since the posterior probability is still of the Gibbs type [17,18], we have to minimize $U(d|g) = U(g|d) + U(d)$, where $U(g|d)$ is the *observation term* and $U(d)$ is the *a-priori term* [17]. The minimization of the functional $U(d|g)$ is performed by a simulated annealing algorithm using Metropolis sampler [19,18] but other, more efficient, techniques can also be used [20].

2.3. The R-SMW model description

In order to define the MRF model, we introduce two random fields D^l and D^r to estimate the left and the right disparity map (output), two random fields G^l and G^r to record the left and the right observed disparity map, and two random field S^l and S^r to record the SSD error.

The observation fields G^l and G^r are filled with the disparity maps computed using the SSD block matching algorithm, taking in turn the left and the right images as the reference one.

The MRF functional is composed by three terms:

- the a-priori term, that imposes a smoothness constraint on the solution;
- the observation term, that describe how the observations are used to produce the solution. This term implements the multiple windows approach of the SMW algorithm;
- the left–right consistency term, that couples the left disparity and the right disparity values and enforces the left–right consistency.

2.3.1. A-priori term

In the R-SMW, the scene is modelled as composed by a set of planes located at different distances to the observer, so that each disparity value corresponds to a plane in scene. Therefore, we employ a *piecewise constant* [8] model, defined as[1]:

$$U(d) = \sum_{i \in I} \sum_{j \in N_i} \overline{\delta}(d_i, d_j) \tag{3}$$

where d_i and d_j are the estimated disparity values (the realization of the field D) and the function $\overline{\delta}(x, y)$ is defined as:

$$\overline{\delta}(x, y) = \begin{cases} 0 & \text{if } x = y \\ 1 & \text{otherwise} \end{cases} \tag{4}$$

This term introduces a regularization constraint, imposing that all pixels assume the same value in a region, thereby smoothing out isolated spikes.

2.3.2. Observation term

The observation model describes how the observations are used to produce the solution, implementing the multiple windows approach of SMW. As an off-centered window for a pixel is the on-centered window for another pixel, the multiple windows technique used by the SMW reduces to assign a given pixel the disparity computed for one of its neighbours with an on-centered window, namely, the neighbour with the smallest SSD error. The nine 7×7 windows scheme produces a sparse neighbourhood of nine pixels. The idea is to generalize and this scheme: the observation term introduces a local non-isotropic relaxation, favouring the neighbour observations with the smallest SSD error

$$U(g, s|d) = \sum_{i \in I} \sum_{j \in N_i \cup \{i\}} \overline{\delta}(d_i, g_j) \cdot \left(\frac{1}{s_j}\right) \tag{5}$$

where g is the observed disparity map, s is the observed SSD values and d is the disparity estimate. g, s and d are, respectively, the realizations of the fields G, S and D.

In this term, the estimated value at site i, d_i, is compared with all its observed neighbours $\{g_j\}_{j \in N_i}$ and with g_i itself. When d_i takes the disparity of one (or more) of its neighbours, one (or more) term(s) in the sum vanishes. The lower the SSD error of the chosen disparity, the higher the cost reduction.

2.3.3. Left–right consistency term

Let d_i^l be the left disparity (i.e., the disparity computed taking the left image as the reference) at site i, and d_i^r the right disparity at site i. The left–right consistency constraint states that: $d_i^l = -d^r_{i+d_i^l}$. The corresponding energy term is:

$$V(d^l, d^r) = \sum_{i \in I} \overline{\delta}\left(d_i^l, -d^r_{i+d_i^l}\right) \tag{6}$$

In this way, we introduce a payload when the left–right constraint is violated.

2.4. Summing up

The final MRF functional is:

$$\begin{aligned} U(d^l, d^r | g^l, s^l, g^r, s^r) = k_1 \cdot [U(g^l, s^l | d^l) + U(g^r, s^r | d^r)] \\ + k_2 \cdot [U(d^l) + U(d^r)] \\ + k_3 \cdot V(d^l, d^r) \end{aligned} \tag{7}$$

where $U(g^l, s^l|d^l)$ and $U(g^r, s^r|d^r)$ are the observation terms applied to the left and right disparity, $U(d^l)$, $U(d^r)$ are the a-priori terms and $V(d^l, d^r)$ is the left–right consistency term. The positive weights k_1, k_2, k_3 are the parameters that control the performance of the algorithm. As the absolute magnitude of the functional is not important in the MRF minimization, we can set $k_1 + k_2 + k_3 = 1$, thereby reducing the free parameters to only two, and delimiting the search space.

3. Tabu search

Tabu search is an optimization method introduced by Fred Glover in 1986 [21]. Other techniques were considered for the given task, like genetic algorithms and particle swarm optimization. Reactive tabu search, described later, was chosen among them because is able to quickly explore a unknown domain without the need of parameter tweaking [6].

Tabu search stems from the research in prohibition-based methods, such as the denial strategy, the reduction strategy

[1] We will omit superscript l and r in the field variables. It is understood that the a-priori model and the observation model applies to both left and right fields.

or even the cutting planes algorithm [22]. The common characteristic of all these algorithms is that they systematically violate feasibility conditions or local optimality boundaries. In particular, local based search and prohibition of moves are the key ideas of the tabu search. Using these rules, tabu search imposes and releases constraints to permit exploration of otherwise forbidden regions [21].

TS is based on the systematic use of memory: it will not only remember the current and best solution but it will also keep information on the itinerary through the last solutions visited. Such information will be used in order to guide the transition from the current to the next solution. In particular, the flexible use of memory embodies the dual processes of creating and exploiting structures for taking advantage of history, combining the activities of acquiring and profiting from information [21].

The overall approach of TS consists of combining a hill-climbing search strategy based on a set of elementary moves and a heuristics to avoid the stops at suboptimal points and the occurrence of cycles [21]. A finite-size list of forbidden moves (i.e., tabu moves) is derived from the recent history of the search in order to forbid or penalize moves towards points already visited. This mechanism mitigates the problem of local minima trapping while driving the search to unexplored regions of the solution space.

The following components define the TS.

Fitness function: This is a scalar function defined over the solution set, that returns a score for each solution.

Move: A move is a procedure by which a new (feasible) solution is generated from the current one.

Neighbourhood: A neighbourhood of a solution is the set of all the solutions that can be reached with one move.

Tabu list: This is a list of moves that are forbidden (or tabu). Its length is fixed but it is updated dynamically with the last move that was picked.

Aspiration conditions: These are rules that overrides tabu restrictions. If the aspiration condition is satisfied, a tabu move becomes allowed. For example, a forbidden move can be selected if the corresponding solution is the best encountered so far. Research has produced many variations on this topic.

Diversification and intensification: Diversification strategies drive the search into new regions. The most common diversification strategy is a random restart, preceded by a step which inserts in the tabu list the most common moves. *Intensification* strategies reinforce move combinations and solution features historically found good. A usual intensification strategy consists in clearing or shortening for a predefined number of iterations the tabu list and restarting from the best solution found.

Therefore, the basic TS algorithm can be described as follows:

(1) Given a starting solution, compute its fitness.
(2) Generate the neighbourhood of the current solution, or, equivalently, a set of candidate moves. A move is allowed if it is not tabu or it satisfies the aspiration condition. Pick the allowed move that get to the best solution and consider it to be the new current solution.
(3) Repeat step 2 until some termination conditions are satisfied.

At each iteration, the chosen move is put in the tabu list, thereby preventing the algorithm to go back to recently visited solutions.

One of the main drawback to solve when dealing with the tabu search algorithm is parameter tuning. Tuning is often needed to obtain competitive results and requires either a deep knowledge of the problem structure or a time consuming and not always reproducible tinkering process. The most critical parameter usually is the tabu list size, which compromises between intensification and diversification strategies. Furthermore, given the fixed size of the tabu list, the search might be trapped in a cycle of length greater than the size list.

In order to cope with this drawback, the *reactive tabu search* (RTS) [6] has been proposed, which dynamically adjusts the tabu list size.

3.1. Reactive tabu search

Reactive tabu search solves the tuning problems by integrating in the tabu search framework a simple history-based feedback scheme for on-line determination of free parameters. In particular, intensification and diversification are automatically balanced to match the local characteristics of the context: the search is intensified in promising regions and diversified when exploring uncharted territories.

In this sense, the feedback scheme can be seen as a simple form of reinforcement learning: the trajectory in the solution space (the history of the search) influences the behavior of the system. This approach effectively shifts the need to gather, analyze and understand the history of the search from the user to the algorithm.

In order to transform a tabu search algorithm in its Reactive counterpart, two modifications are needed. The first consists in introducing a *feedback* scheme into the method. This is usually done by adjusting the tenure of the tabu list during the search. The second is the insertion of a *escape strategy*, which adds another powerful defense line against local minima trapping.

3.1.1. Feedback schemes

They are usually implemented by dynamically adjusting the tabu list size according to moves presented in the tabu list. The need of diversification, which corresponds to a tenure increase, is triggered by the repetition of previously visited configurations. On the other hand, tenure decreases when this need disappears.

In order to increase the efficiency in handling repetition checks, radix-tree or hashing are commonly implemented. This last approach could lead to erroneously forbidden moves, but as long as there is an aspiration criteria in place and the collision probability is sufficiently low, the influence on results can be proven negligible [6].

3.1.2. Escape strategy

It introduces a more radical diversification in order to increase the robustness of the algorithm. The escape phase is triggered when too many configurations are repeated too often. In particular this phase prevents the so called *chaotic trapping* (i.e., when the solution chain is limited in a finite portion of the search space but non-periodical).

A simple escape strategy consists of a number of random steps executed starting from the current configuration (possibly with a bias toward steps that bring the trajectory away from the current search region).

4. RTS applied

In this section we specify how the reactive tabu search framework is applied to our specific parameter optimization problem. The R-SMW algorithm, described in Section 2, has two independent variables to be estimated, the weights k_1 and k_2 (the third one, k_3, is always defined as $k_3 = 1 - k_1 - k_2$). Coupled with the positivity constraint, this restricts the search space to a closed region of the plane. In the following, we are going to define the components that constitute a complete RTS framework.

4.1. Solution and move

A *solution* is a point in the region of the plane k_1, k_2 limited by the axes and by the line $k_2 = 1 - k_1$ (*the search space*). A *move* consists in changing the value of one parameter[2] k_i by choosing the new value among its nearest neighbours on a coarse-to-fine multi-resolution discretization of the chosen axis (this is achieved by flipping one bit of its fixed point binary representation) and adjusting the other two in such a way that their ratio is kept the same. This choice solves in a natural way the problem of being able to move to a far (in a euclidean sense) yet related solution and guarantees against minima trapping in a noisy and arbitrarily complex search space. This also gives three directions along which one can move starting from the current solution.

4.2. Fitness function

It measures the performance of the R-SMW stereo algorithm as the difference between the estimated disparity and the ground truth. The independent variables of the fitness function are the weights k_1 and k_2. In particular, the computation of the fitness function proceeds as follows:

- given a solution (parameter set) $s^\ell = (k_1^\ell, k_2^\ell)$,
- run the stereo process with s^ℓ and find the disparity map D^ℓ,
- compute the fitness $f(s^\ell) = -\mathrm{err}(D^\ell, D^o)$ where D^o is the ground truth disparity.

Following [7], the disparity error is given by the fraction of wrong matches in non-occluded regions:

$$\mathrm{err}(D^\ell, D^o) = \frac{1}{N} \sum_{(i,j)\in I\setminus B} \overline{\delta}(D^\ell(i,j), D^o(i,j)) \tag{8}$$

where N is the number of pixels, B is the set of occluded pixels (provided with the ground truth), and $\overline{\delta}(x,y)$ has been defined in Eq. (4).

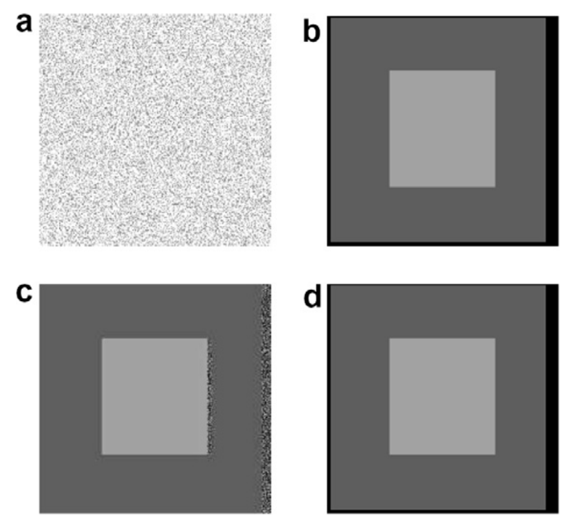

Fig. 1. Random Dots Stereograms: a square. (a) Left image; (b) disparity obtained by SMW; (c) disparity obtained by R-SMW with optimal parameters; (d) ground truth disparity. Both SMW and R-SMW achieves zero disparity error.

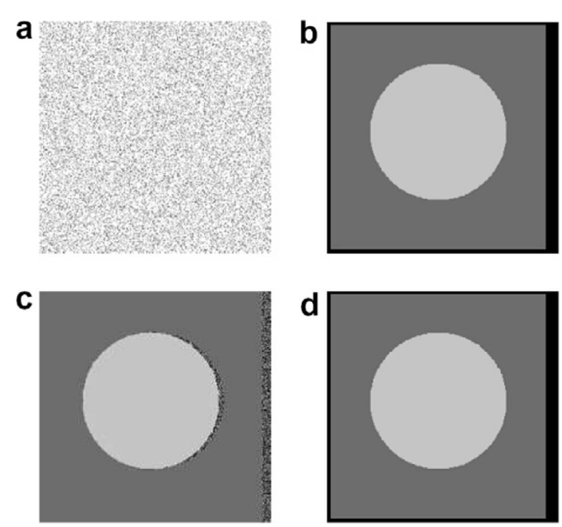

Fig. 2. Random Dots Stereograms: a circle. (a) Left image; (b) disparity obtained by SMW; (c) disparity obtained by R-SMW with optimal parameters; (d) ground truth disparity. Both SMW and R-SMW achieves zero disparity error.

[2] We can virtually change k_3 by operating on k_1 and k_2.

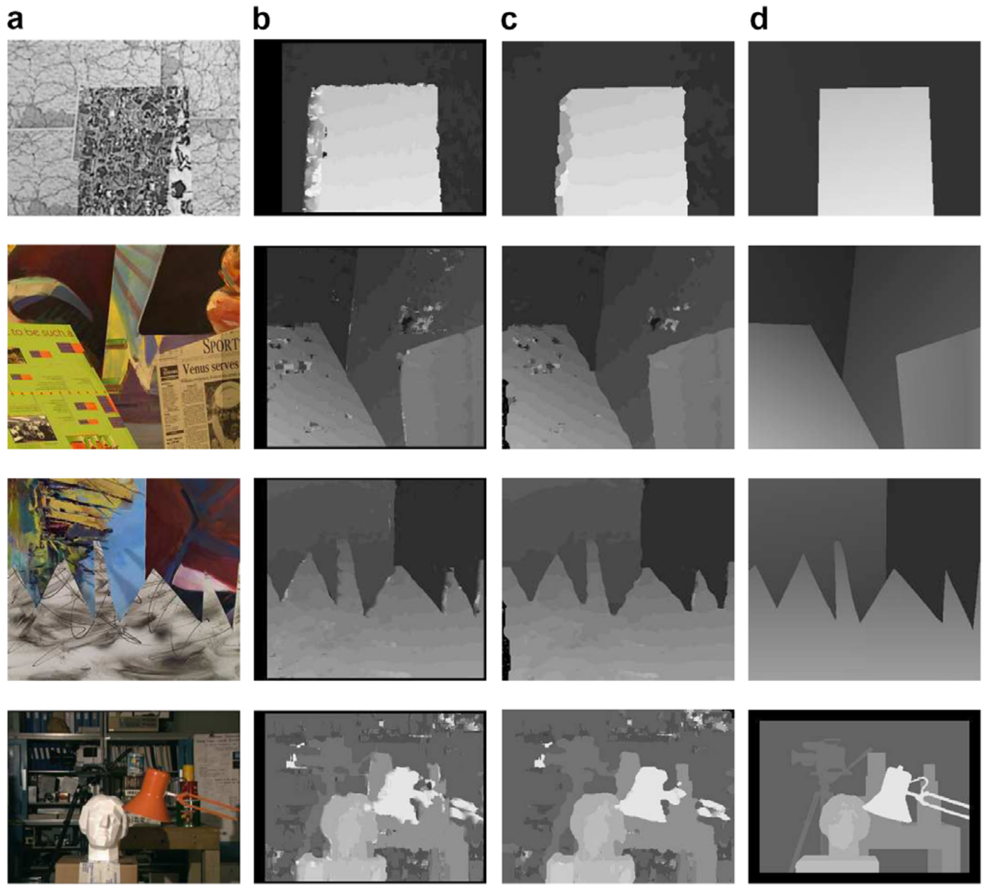

Fig. 3. Experiments with real data, Middlebury set. (a) Left image of the stereo pair; (b) disparity obtained by SMW; (c) disparity obtained by R-SMW with optimal parameters; (d) ground truth disparity. Each row corresponds to a different stereo pair: (from top to bottom) Map, Venus, Sawtooth, and Tsukuba.

Table 1
Optimal parameters and disparity errors for each stereo pair

Stereo pair	Parameters	Err
Map	(0.03, 0.88, 0.09)	0.26
Venus	(0.67, 0.26, 0.07)	2.92
Sawtooth	(0.69, 0.21, 0.10)	2.38
Tsukuba	(0.56, 0.12, 0.32)	4.67

Table 2
Joint optimal parameters and disparity errors for each stereo pair

Stereo pair	Parameters	Err
Map	(0.68, 0.23, 0.09)	0.55
Venus	(0.68, 0.23, 0.09)	3.16
Sawtooth	(0.68, 0.23, 0.09)	2.41
Tsukuba	(0.68, 0.23, 0.09)	4.71
Total	(0.68, 0.23, 0.09)	10.83

4.3. Tabu list and aspiration conditions

The *tabu list* is always updated with the last chosen move. The *aspiration condition* says that if a move leads to a better solution it is chosen even if is tabu. According to the *reactive* paradigm, the size of the tabu list is increased when configurations are repeated, otherwise it is reduced. More in detail, for each iteration if the value of the fitness function increases, the size of the tabu list is reduced thus favouring the intensification of the search. Otherwise the length of the tabu list is increased, promoting diversification.

Table 3
Optimal parameters and errors for the planar stereo pairs

Stereo pair	Parameters	Err	L1O err
Barn1	(0.69, 0.20, 0.11)	1.69	1.69
Barn2	(0.67, 0.19, 0.14)	2.87	2.88
Bull	(0.68, 0.22, 0.10)	2.42	2.42
Poster	(0.64, 0.25, 0.11)	3.15	3.18
Sawtooth	(0.69, 0.21, 0.10)	2.38	2.39
Venus	(0.67, 0.26, 0.07)	2.92	2.93

The err column shows the errors obtained using the optimal parameters for each couple. The L1O err column displays the errors calculated using the parameters from the joint optimization of all the sets but the current.

4.4. Escape strategy

The *escape* mechanism is defined by carrying out a random restart whenever the fitness function has not increased after a fixed number of iterations.

The initial solution gives all parameters equal magnitude: however, we verified empirically that the performance of the method is not affected by the starting point. The search ends when for a given number of iterations the best solution found has not been updated.

5. Experiments

In this section experiments are reported for both synthetic and real cases.

First we estimated the optimal parameters for Random Dots Stereograms (RDS). The fitness function was com-

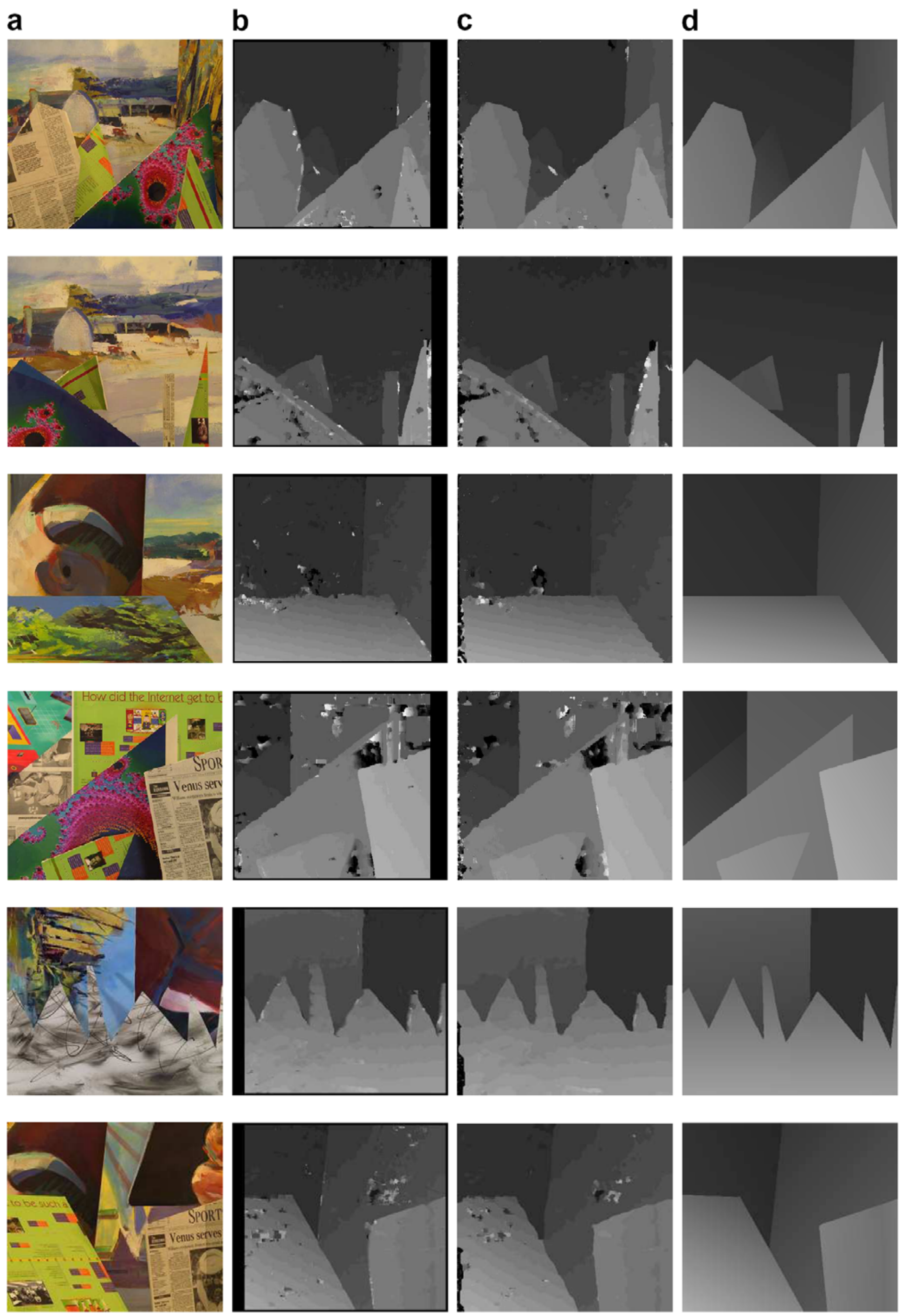

Fig. 4. Experiments with real data, planar sets. (a) Left image of the stereo pair; (b) disparity obtained by SMW; (c) disparity obtained by R-SMW with optimal parameters; (d) ground truth disparity. Each row corresponds to a different stereo pair: (from top to bottom) Map, Venus, Sawtooth, and Tsukuba.

puted using a square RDS and circular RDS (Figs. 1 and 2). The RTS optimization found the following values for the parameters $k_1 = 1$ (the *observation* weight), $k_2 = k_3 = 0$ (the *a priori* and *L–R* consistency weights, respectively), which reproduced the behaviour of the original SMW algorithm (not considering occluded areas), as can be seen in Fig. 1.

These values of the parameters make very sense: the a-priori term is not needed since there is no noise; the left–right consistency term is useless as well, because the two synthetic views are perfectly consistent in the non-occluded areas.

Then, we carried out experiments with the Middlebury data set [7]: it consists of four stereo pairs: Map, Venus, Sawtooth and Tsukuba (Fig. 3). We estimated the optimal parameters separately for each stereo pair. They are reported in Table 1, together with the disparity errors computed according to Eq. (8). Fig. 3 shows the results obtained with these parameters for each stereo pair.

In order to assess the sensitivity of the parameters to the training set, we also carried out the estimation process using all the four pairs, defining the fitness function as the sum of the fitness for each set (we call this the *joint* fitness function). The parameters yielded by the joint estimation and the disparity errors achieved by each stereo pair are shown in Table 2. The parameters estimated with both the single and joint fitness outperformed the manually tuned parameters previously used [3].

It is worth noting that Sawtooth and Venus images are similar (they are both composed of well-textured slanted planes) and so are the optimal parameters computed for these two stereo pairs. This seems to suggest that there are optimal parameters for classes of similar images. To verify this intuition, a further set of images very similar to Sawtooth and Venus has been considered (Barn 1, Barn 2, Poster and Bull from the Middlebury data set [7]). The estimated parameters are listed in Table 3 and the corresponding disparity maps in Fig. 4. As we expected, the results obtained are indeed remarkably close to the ones estimated for Sawtooth and Venus stereo pairs.

This means that once the optimal set of parameters is known for a certain image it can be applied to new images sharing similar characteristics without knowing the ground truth or performing any kind of parameter search.

To definitely confirm the feasibility of this approach we performed a leave-1-out test [23] on the complete data-set of planar stereo couples, in which the fitness of a pair was calculated with the parameters obtained from the joint optimization of the other ones. The results are shown in the last column of Table 3.

Experiments were carried out on various Pentium 4 class machines: the average time for a single sampling of the fitness function was about forty seconds. The number of samplings required to exit the optimization process for our experiments was typically of a few hundred points; producing results of comparable precision by exhaustive search would have required several thousands of evaluations.

6. Conclusion

The purpose of this paper has been to show that MRF parameter tuning can be automated by using an optimization strategy. We concentrated on stereo matching with a MRF-based algorithm (R-SMW) and used reactive tabu search for parameters optimization. The core ingredient is the fitness function, that measures the performance of a particular parameters set. The usefulness of such an approach is based on the fact that there are optimal parameters that are valid for classes of images, instead of being image-specific. So, once the optimal parameters have been obtained from a representative of a class of images, its parameters can be used without the need of further processing. This is what is needed in most of the real-world tasks.

References

[1] Y. Boykov, O. Veksler, R. Zabih, Fast approximate energy minimization via graph cuts, in: Proceedings of the Seventh IEEE International Conference on Computer Vision, 1999, pp. 377–384.
[2] P. Felzenszwalb, D. Huttenlocher, Efficient belief propagation for early vision, in: International Conference on Computer Vision and Pattern Recognition.
[3] A. Fusiello, U. Castellani, V. Murino, Relaxing symmetric multiple windows stereo using markov random fields, in: M. Figureido, J. Zerubia (Eds.), Energy Minimization Methods in Computer Vision and Pattern Recognition, Lecture Notes in Computer Science, Vol. 2124, Springer, 2001, pp. 91–104.
[4] A. Fusiello, V. Roberto, E. Trucco, Efficient stereo with multiple windowing, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society Press, Puerto Rico, 1997, pp. 858–863.
[5] F. Glover, M. Laguna, Tabu Search, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998.
[6] R. Battiti, G. Tecchiolli, The reactive tabu search, ORSA Journal on Computing 6 (2) (1994) 126–140.
[7] D. Scharstein, R. Szeliski, A taxonomy and evaluation of dense two-frame stereo correspondences algorithms, International Journal of Computer Vision 1 (47) (2002) 7–42.
[8] M.Z. Brown, D. Burschka, G.D. Hager, Advances in computational stereo, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (8) (2003) 933–1008.
[9] L. Cinque, S. Levialdi, G. Pignalberi, R. Cucchiara, S. Martinz, Optimal range segmentation parameters through genetic algorithms, in: Proceedings of the International Conference on Pattern Recognition, 2000, pp. 1474–1477.
[10] G. Pronzato, A. Wallace, Adaptive control of a boundary detection algorithm, in: Proc. of IEEE International Conference on Image Processing and its Applications, Dublin, 1997, pp. 356–360.
[11] P. Fua, Combining stereo and monocular information to compute dense depth maps that preserve depth discontinuities, in: Proceedings of the International Joint Conference on Artificial Intelligence, Sydney, Australia, 1991, pp. 1292–1298.
[12] O. Faugeras, B. Hotz, H. Mathieu, T. Viéville, Z. Zhang, P. Fua, E. Théron, L. Moll, G. Berry, J. Vuillemin, P. Bertin, C. Proy, Real-time correlation-based stereo: algorithm, implementation and applications, Tech. Rep. 2013, Unité de recherche INRIA Sophia-Antipolis (August 1993).
[13] D. Geiger, B. Ladendorf, A. Yuille, Occlusions and binocular stereo, International Journal of Computer Vision 14 (3) (1995) 211–226.
[14] I.J. Cox, S. Hingorani, B.M. Maggs, S.B. Rao, A maximum likelihood stereo algorithm, Computer Vision and Image Understanding 63 (3) (1996) 542–567.

[15] T. Kanade, M. Okutomi, A stereo matching algorithm with an adaptive window: theory and experiments, IEEE Transactions on Pattern Analysis and Machine Intelligence 16 (9) (1994) 920–932.
[16] S.S. Intille, A.F. Bobick, Disparity-space images and large occlusion stereo, in: J.-O. Eklundh (Ed.), European Conference on Computer Vision, Springer-Verlag, Stockholm, Sweden, 1994, pp. 179–186.
[17] S. Geman, D. Geman, Stochastic relaxation, gibbs distribution, and bayesian restoration of images, IEEE Transactions on Pattern Analysis and Machine Intelligence 6 (6) (1984) 721–741.
[18] J.K. Marroquine, Probabilistic solution of inverse problem, Ph.D. thesis, Massachusetts Institute of Technology (1985).
[19] C.G.J.S. Kirkpatrik, M. Vecchi, Optimization by simulated annealing, Science 220 (4) (1983) 671–680.
[20] S.Z. Li, Markov Random Field Modeling in Computer Vision, Computer Science Workbench, Springer, Tokyo, 1995.
[21] F. Glover, Future paths for integer programming and links to artificial intelligence, Computers and Operations Research 13 (1986) 533–549.
[22] A. Hertz, E. Taillard, D. de Werra, A tutorial on tabu search, Tech. rep., EPFL, Départment de Mathétiques, MA-Ecublens, CH-1015, Lausanne (1995).
[23] D.G.S.R.O. Duda, P.E. Hart, Pattern Classification, second ed., John Wiley and Sons, New York, 2001.