Reg. Num. 372-032

A quadratic programming procedure for elastic image matching

V.V. Mottl, A.V. Kopylov, O.V. Krasotkina

Tula State University, Lenin Ave. 92, 300600 Tula, Russia
Phone: 087-2 35-36-37. Fax: 087-2 33-13-05. E-mail: ko180177@uic.tula.ru, kopylov@uic.tula.ru, vmottl@yahoo.com

ABSTRACT

Nonlinear transformation of one image plane relative to another by spatially constrained elastic matching of two pixel grids is proposed as a technique of measuring image similarity for the purpose of stereo image analysis, face and fingerprint identification. The elastic matching algorithm is devised as a combination of two pair-wise separable quadratic programming procedures applied independently to each row and then to each column of the pixel grid. The method provides the linear computational complexity with respect to the number of pixels without application of parallel computers.

KEYWORDS: image analysis, image matching, quadratic programming, dynamic programming.

1. INTRODUCTION

Despite the dramatic variety of application problems of image analysis, it is possible to set off some single subclasses of problems which allow for treating them in unified terms of respective standard mathematical optimization problems for which there exist effective methods of solving.

In this paper, we restrict our concern to the one subclass of image processing problems of that sort, in which it is required to find a pair-wise relation between elements of two images, i.e. image matching. As examples of such problems, at least, three of them should be mentioned, namely, the problem of stereo image analysis, face identification, fingerprints pattern type recognition. In these problems, image matching would allow to build a decision rule for face and fingerprints pattern type identification by comparing the given image with a number of basic images, and disparity map for stereo image analysis.

In the image matching procedure, a pair of images of the same structure is analyzed, for instance, two human portraits, two fingerprints, or a stereo pair of photographs of the same object taken from two slightly different directions (Fig. 1). Each of the images forming a pair, let it conventionally be the left $\mathbf{y}^l = (y_\mathbf{i}^l, \mathbf{i} \in \mathbf{I})$ and the right image $\mathbf{y}^r = (y_\mathbf{i}^r, \mathbf{i} \in \mathbf{I})$, is a real-valued gray-level vector on a rectangular pixel grid $\mathbf{I} = \{\mathbf{i} = (i_1, i_2),\ i_1 = 1,\ldots,n_1,\ \ i_2 = 1,\ldots,n_2\}$, and the image pair as a whole can be considered as a two-component array. It is required to establish a correspondence in the two images between the pairs of points which could be treated as identical.

However, immediate comparison of the vectors formed by the original pixel grids of two images is senseless because of inevitable distortions caused by different registration conditions, so that the identical coordinates $y_\mathbf{i}^l$ and $y_\mathbf{i}^r$ of the two gray-level vectors $\mathbf{y}^l$ and $\mathbf{y}^r$ will correspond to incomparable points of the images.

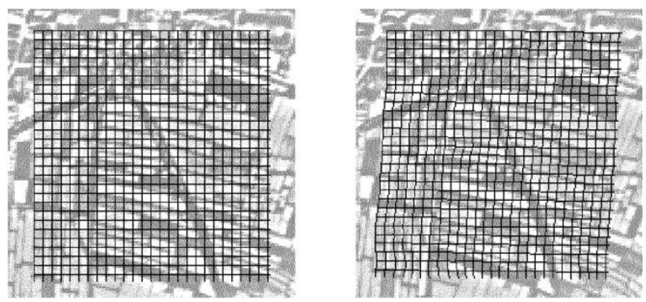


Figure 1. An example of image matching problem: two satellite photographs forming a stereo pair and disparity map of their identical points.

A widely adopted way to overcoming this obstacle is elastic image matching [1]. The main and most difficult step of elastic matching is building so-called disparity map which for each point $\mathbf{i}$ of the one of the images (conventionally taken as the reference image) determines a vector $\mathbf{x_i} = (x_\mathbf{i}^1, x_\mathbf{i}^2) \in R^2$ indicating the spatial shift between $\mathbf{i}$ and its match $\mathbf{i}' = \mathbf{i} + \mathbf{x_t}$ in the second (search) image with eventual interpixel interpolation $\mathbf{I}' = (\mathbf{i}', \mathbf{i}' \in R^2)$ [2]. Thus the result of matching two images $\mathbf{y}^l$ and $\mathbf{y}^r$ should be represented as a secondary vector data field (disparity map) $\mathbf{X} = (\mathbf{x_i}, \mathbf{i} \in \mathbf{I})$ with the same range of definition as the reference image.

We consider here only a criterion of disparity map building and one of possible algorithms of image matching. As to the application of this approach to stereo image analysis, face and fingerprints identification, the respective technique and experimental results will be discussed elsewhere.

2. THE BASIC CRITERION OF ELASTIC IMAGE MATCHING

Let us seek the desired disparity map $\mathbf{X}=(\mathbf{x_i},\, \mathbf{i}\in\mathbf{I})$ in the form of a displacement vector field, utilizing two kinds of information about the initial pair of images.

On the one hand, the sought vector field $\mathbf{X}$ should provide as good agreement between two images as possible. It is convenient to express such a judgment in the form of a quadratic function

$$\varphi_{\mathbf{i}}(\mathbf{x_i}\,|\,y^l_{\mathbf{i}},y^r_{\mathbf{i}}) = b^0 + (\mathbf{x_i}-\mathbf{x}^0_{\mathbf{i}})^T\,\mathbf{Q}^{\mathbf{0}}_{\mathbf{i}}(\mathbf{x_i}-\mathbf{x}^0_{\mathbf{i}})^T \cong (y^l_{\mathbf{i}}-y^r_{\mathbf{i+x(i)}})^2 \to \min, \qquad (1)$$

where $y^l_{\mathbf{i}}$ and $y^r_{\mathbf{i+x(i)}}$, are normalized brightness values or appropriate local features of the image point. Let such a function be called image-dependent function.

On the other hand, the sought vector field of transformation would be smooth enough, as it provides matching of two images of the same structure. This knowledge can be expressed as a quadratic function

$$\gamma_{\mathbf{i,j}}(\mathbf{x_i},\mathbf{x_j}) = \|\mathbf{x_i}-\mathbf{x_j}\|^2 = (\mathbf{x_i}-\mathbf{x_j})^T\,\mathbf{U}_{\mathbf{i,j}}(\mathbf{x_i}-\mathbf{x_j}) \to \min,$$
$$(\mathbf{x_i},\mathbf{x_j}\in R^2,\ \mathbf{i},\mathbf{j}\in\mathbf{I}), \qquad (2)$$

which is defined over $R^2\times R^2$ at each pair of neighboring pixels. Let such a function be called model function.

The pixel neighborhood can be expressed by an lattice-like undirected graph G which is understood as a set of pixel pairs associated with edges (Fig. 2).

Therefore, the image-dependent function (1) may be called also node function, since it is a function of a single variable associated with a node of graph G. Model function (2), on the contrary, depends of two variables associated with an edge of the graph and, so, may be denominated as edge function.

Finding a compromise between two mutually contradictory requirements (1) and (2) is achieved by optimizing a combined quadratic criterion

$$J(\mathbf{X}\,|\,\mathbf{y}^l,\mathbf{y}^r) = \sum_{\mathbf{i}\in\mathbf{I}}\varphi_{\mathbf{i}}(\mathbf{x_i}\,|\,y^l_{\mathbf{i}},y^r_{\mathbf{i}}) + \sum_{(\mathbf{i},\mathbf{j})\in G}\gamma(\mathbf{x_i},\mathbf{x_j}). \qquad (3)$$

The number of its variables is equal to the number $|\mathbf{I}|$ of nodes in graph G, i.e. the number of pixels in the reference image. We call a criterion of such a kind pair-wise separable objective function [3,4], because it is sum of partial criteria each of which is function of not more than two variable.

We consider the desired disparity map as a displacement vector field that provides the minimum value of the basic criterion (1)

$$\hat{\mathbf{X}} = \begin{pmatrix} \mathbf{x}_{11} & \cdots & \mathbf{x}_{n_1 1} \\ \cdots & \cdots & \cdots \\ \mathbf{x}_{1 n_2} & \cdots & \mathbf{x}_{n_1 n_2} \end{pmatrix} = \arg\min J(\mathbf{X}\,|\,\mathbf{y}^l,\mathbf{y}^r). \qquad (4)$$

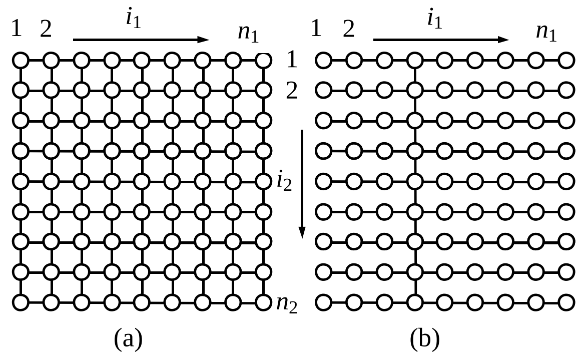


Figure 2. Neighborhood graphs on the pixel grid: G - rectangular lattice (a) and T - simplest tree (b).

Furthermore, the desired vector field is to be chosen so that there are no “folds” in the image plane of the reference image (Fig. 3).

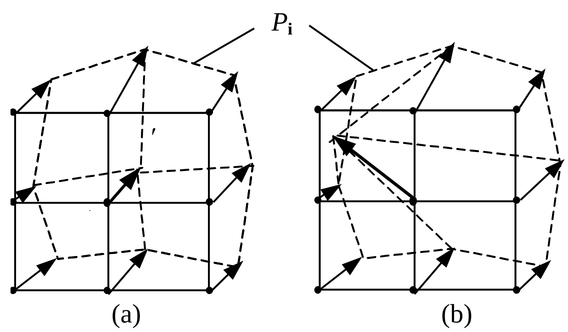


Figure 3. A disparity map of two images: without “folds” (a) and with “folds” (b).

To prevent the folds, each spatial shift vector $\mathbf{x_i}$ is to be chosen so that the match point $\mathbf{i}'=\mathbf{i}+\mathbf{x_i}$ in the second image would get inside the polygon $P_{\mathbf{i}}$ (Fig. 3,a). This means, that the admitted region $\mathcal{X}_{\mathbf{i}}$ of sought vector $\mathbf{x_i}$ is defined as follows:

$$\mathcal{X}_{\mathbf{i}} = \{\mathbf{x_i} : \mathbf{i}'\in P_{\mathbf{i}}\},\ \mathbf{i}'=\mathbf{i}+\mathbf{x_i}.$$

In this case, the minimization of the criterion (3) must be carried out with respect to the constraints noticed above:

$$\hat{\mathbf{X}} = \arg\min_{\mathbf{x_i}\in\mathcal{X}_{\mathbf{i}},\,\mathbf{i}\in\mathbf{I}} J(\mathbf{X}\,|\,\mathbf{y}^l,\mathbf{y}^r). \qquad (5)$$

The problem (5) is a problem of constrained quadratic optimization. The methods of solving problems of such a kind (simplex method, interior-point method) [5] are iterative and their computational complexity is proportional to squared number of pixels of the original pixel-grid $\mathbf{I}$.

However, the criterion (5) is G-supported separable goal function with constraints that are not pair-wise separable with respect to the neighborhood graph G, because each constraint joints eight points of $\mathbf{I}$, that not all are the immediate neighbor of node $\mathbf{i}$. Unfortunately, the intent to create a non-iterative procedure for finding the minimum point of a pair-wise separable function (3) with variable adjacency graph of general kind under non-separable constraints comes up against a fundamental barrier.

Procedures like stochastic relaxation and simulated annealing [6] improve the exhaustive search to some extent but remain extremely time-consuming.

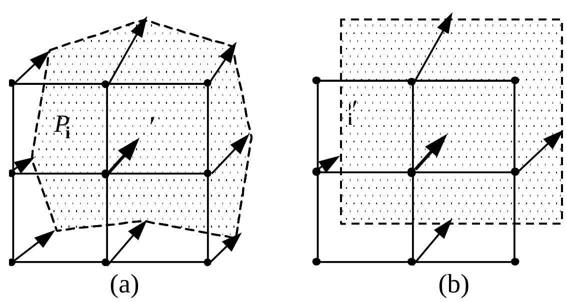
But in the particular case, when the constraints are separable and the variable-adjacency graph G has no cycles, i.e. is a tree, it is possible to construct a highly effective global optimization procedure based on a recurrent decomposition of the initial problem of minimizing the pair–wise separable function (3) of $|\mathbf{I}|$ variables $\mathbf{X}=(\mathbf{x_i},\, \mathbf{i}\in\mathbf{I})$ into a succession of $|\mathbf{I}|$ partial problems, each of which consists in constrained minimizing a function of only one variable $\mathbf{x_i}$. The case of non-constrained minimization of goal functions with tree-like separability is considered in [3, 4].

3. PARTIAL CRITERIA FOR SINGLE COLUMNS: A SERIES OF TREES INSTEAD OF THE PIXEL-NEIGHBORHOOD LATTICE

To utilize the advantages of the separable constraints and the tree as variable-adjacency graph, we resort here to the following two heuristic tricks. First, we replace the admitted region $P_{\mathbf{i}}$ of variable $\mathbf{x_i}$ by a rectangle $P'_{\mathbf{i}}$, so that the constraints become more hard but separable (Fig. 4).

In this case, for each vector $\mathbf{x_i}\in\mathcal{X}_{\mathbf{i}}\subseteq R^2$ of the sought disparity map $\mathbf{X}=(\mathbf{x_i},\, \mathbf{i}\in\mathbf{I})$, the admitted region is

$$\mathcal{X}_{\mathbf{i}}=\left\{\mathbf{x_i}:\ g(\mathbf{x_i},\mathbf{x_j})\geq 0, (\mathbf{i},\mathbf{j})\in G\right\},\quad g(\mathbf{x_i},\mathbf{x_j})=\mathbf{a}_{\mathbf{j}}^T\left(\mathbf{x_i}-\mathbf{x_j}\right)-1.$$



(a) (b)

Figure 4. The admissible region of variable $\mathbf{x_i}$ in the case of non-separable constraints (a) and separable constraints (b).

Second, instead of minimizing the full-lattice objective function (3), (5), we minimize a series of partial tree-like ones $J^{(T,i_1)}(\mathbf{X}\,|\,\mathbf{y}^l,\mathbf{y}^r)$, $i_1=1,\dots,n_1$, each of the same full set of variables $\mathbf{X}=(\mathbf{x_i},\, \mathbf{i}\in\mathbf{I}),\ \mathbf{x_i}\in\mathcal{X}_{\mathbf{i}}$ but obtained from (3) by removing the edge functions corresponding to vertical edges of the lattice except one column (Fig. 2,b):

$$J^{(T,i_1)}(\mathbf{X}\,|\,\mathbf{y}^l,\mathbf{y}^r)=\sum_{i_1=1}^{n_1}\sum_{i_2=1}^{n_2}\varphi_{i_1,i_2}\left(\mathbf{x}_{i_1,i_2}\middle|\mathbf{y}^l_{i_1,i_2},\mathbf{y}^r_{i_1,i_2}\right)+\sum_{i_1=2}^{n_1}\sum_{i_2=2}^{n_2}\gamma\left(\mathbf{x}_{i_1-1,i_2},\mathbf{x}_{i_1,i_2}\right)+\sum_{i_2=2}^{n_2}\gamma\left(\mathbf{x}_{i_1,i_2-1},\mathbf{x}_{i_1,i_2}\right)\to\min_{\substack{g(\mathbf{x_i},\mathbf{x_j})\geq 0,\\ (\mathbf{i},\mathbf{j})\in T}},\quad (6)$$

where upper index (T,i_1) indicates that adjacency of variables in this criterion is determined by the tree T with the i_1 th column as the stem.

Let $\hat{\mathbf{X}}^{(T,i_1)}=(\hat{\mathbf{x}}^{(T,i_1)}_{i_1,i_2},\ i_1=1,\dots,n_1,\ i_2=1,\dots,n_2)$ be the vector field obtained as the minimum point of the criterion (6) for the i_1 th stem

$$\hat{\mathbf{X}}^{(T,i_1)}=\arg\min J^{(T,i_1)}(\mathbf{X}\,|\,\mathbf{y}^l,\mathbf{y}^r),$$

and $(\hat{\mathbf{x}}^{(T,i_1)}_{i_1,1},\dots,\hat{\mathbf{x}}^{(T,i_1)}_{i_1,n_2})$ be the i_1 th column cut out of it. The main heuristic idea of reducing calculations when forming the displacement-vector field of elastic transformation is to assemble the displacement vector field from these partially optimal stem-columns

$$\hat{\mathbf{X}}^{(T,i_1)}=\begin{pmatrix}\hat{\mathbf{x}}^{(T,1)}_{1,1} & \cdots & \hat{\mathbf{x}}^{(T,n_1)}_{n_1,1}\\ \cdots & \cdots & \cdots\\ \hat{\mathbf{x}}^{(T,1)}_{1,n_2} & \cdots & \hat{\mathbf{x}}^{(T,n_1)}_{n_1,n_2}\end{pmatrix}.\quad (7)$$

4. A QUADRATIC PROGRAMMING PROCEDURE FOR OPTIMIZATION OF A SEPARABLE FUNCTION SUPPORTED BY A CHAIN

In this section, we consider an optimization procedure for objective function with chain-like variable adjacency graph. As it is shown in the next section, this procedure is fundamental for elastic matching.

We don’t associate here the nodes of the supporting chain with pixels of an image. Thus, the nodes are not associated with points of a plane $\mathbf{i}=(i_1,i_2)$, therefore, to denote them, we will use symbol i instead of $\mathbf{i}$ in order to avoid any resemblance to vectors.

Let G be a chain with the set of nodes I (Fig. 5), and

$$J(\mathbf{X})=\sum_{i=1}^{n}\varphi_i(\mathbf{x}_i)+\sum_{i=2}^{n}\gamma_i(\mathbf{x}_{i-1},\mathbf{x}_i)\quad (8)$$

be a G-supported separable goal function to be minimized in an arbitrary range of definition $\mathbf{x}_i\in\mathcal{X}_i$. Both node and edge functions occurring in the full separable objective function (8) are taken in the quadratic form

$$\varphi_i(\mathbf{x}_i)=b^0+(\mathbf{x}_i-\mathbf{x}_i^0)^T\mathbf{Q}_i^{\mathbf{0}}(\mathbf{x}_i-\mathbf{x}_i^0)^T,$$

$$\gamma_{i-1,i}\left(\mathbf{x}_{i-1},\mathbf{x}_i\right)=(\mathbf{x}_{i-1}-\mathbf{x}_i)^T\mathbf{U}_{i-1,i}(\mathbf{x}_{i-1}-\mathbf{x}_i).$$

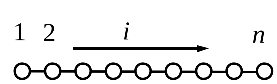


Figure 5. Structure of an arbitrary chain.

Let us suppose first that no constraints are imposed upon the variables $\mathbf{x}_i$.

The natural way of optimization of a separable goal functions (8) is dynamic programming procedure "forward against forward", proposed in [7]. The principal idea of that procedure consists in the following. Let us define, in addition to the chain-like objective function (8), two truncated versions of it, namely the left-side $J_i^-(\mathbf{x}_1,\dots,\mathbf{x}_i)$ and the right-side $J_i^+(\mathbf{x}_i,\dots,\mathbf{x}_n)$ ones, which differ from (8) by that the summation in both sums is done, respectively, only up to the i-th summand and starting from it. Functions

$$\tilde{J}_i^-(\mathbf{x}_i) = \min_{\mathbf{x}_1,\dots,\mathbf{x}_{i-1}} J_i^-(\mathbf{x}_1,\dots,\mathbf{x}_i),$$
$$\tilde{J}_i^+(\mathbf{x}_i) = \min_{\mathbf{x}_{i+1},\dots,\mathbf{x}_n} J_i^+(\mathbf{x}_i,\dots,\mathbf{x}_n)$$

are called, respectively, the left-side and the right-side Bellman functions. Both the left-side and the right-side Bellman functions can be easily found by the following recurrent relations:

$$\begin{aligned}\tilde{J}_i^-(\mathbf{x}_i) &= \varphi_i(\mathbf{x}_i) + \min_{\mathbf{x}_{i-1}}[\gamma_i(\mathbf{x}_{i-1},\mathbf{x}_i) + \tilde{J}_{i-1}^-(\mathbf{x}_{i-1})],\\ \tilde{J}_i^+(\mathbf{x}_i) &= \varphi_i(\mathbf{x}_i) + \min_{\mathbf{x}_{i+1}}[\gamma_i(\mathbf{x}_{i+1},\mathbf{x}_i) + \tilde{J}_{i+1}^-(\mathbf{x}_{i+1})]\end{aligned} \qquad (9)$$

starting from $\tilde{J}_1^-(\mathbf{x}_1) = \varphi_1(\mathbf{x}_1)$ and $\tilde{J}_n^+(\mathbf{x}_n) = \varphi_n(\mathbf{x}_n)$ respectively. Since the criterion (8) is quadratic, Bellman functions (9) are also quadratic

$$\tilde{J}_i^-(\mathbf{x}_i) = \tilde{b}_i^- + (\mathbf{x}_i - \tilde{\mathbf{x}}_i^-)^T \tilde{\mathbf{Q}}_i^- (\mathbf{x}_i - \tilde{\mathbf{x}}_i^-),$$
$$\tilde{J}_i^+(\mathbf{x}_i) = \tilde{b}_i^+ + (\mathbf{x}_i - \tilde{\mathbf{x}}_i^+)^T \tilde{\mathbf{Q}}_i^+ (\mathbf{x}_i - \tilde{\mathbf{x}}_i^+)$$

and defined by recurrent recalculation of their parameters $(\tilde{b}_i^-, \tilde{\mathbf{x}}_i^-, \tilde{\mathbf{Q}}_i^-)$ and $(\tilde{b}_i^+, \tilde{\mathbf{x}}_i^+, \tilde{\mathbf{Q}}_i^+)$.

At any point i, the left-side and the right-side Bellman functions give the so-called marginal function $\hat{J}_i(\mathbf{x}_i) = \min_{\mathbf{x}_s,\, s\neq i} J(\mathbf{x}_1,\dots,\mathbf{x}_n)$, and its minimum immediately specifies the optimal value of the objective variable at this point:

$$\begin{aligned}\hat{J}_i(\mathbf{x}_i) &= \tilde{J}_i^-(\mathbf{x}_i) + \tilde{J}_i^+(\mathbf{x}_i) - \varphi_i(\mathbf{x}_i),\\ \hat{\mathbf{x}}_i &= \arg\min_{\mathbf{x}_i} \hat{J}_i(\mathbf{x}_i).\end{aligned} \qquad (10)$$

In the section 5, the marginal node functions will help us to solve a succession of optimization problems for pixel neighborhood trees (Fig. 2,b) by using a unified procedure.

It can be easily shown [7] that the marginal functions are also quadratic $\hat{J}_i(\mathbf{x}_i) = \hat{b}_i + (\mathbf{x}_i - \hat{\mathbf{x}}_i)^T \hat{\mathbf{Q}}_i (\mathbf{x}_i - \hat{\mathbf{x}}_i)$, and their minimum points can be calculated in the following way:

$$\begin{aligned}\hat{\mathbf{x}}_i &= \hat{\mathbf{Q}}_i^{-1}[\mathbf{U}_i(\mathbf{U}_i + \tilde{\mathbf{Q}}_{i-1}^-)^{-1}\tilde{\mathbf{Q}}_{i-1}^-\tilde{\mathbf{x}}_{i-1}^- + \mathbf{Q}_i^0\mathbf{x}_i^0 +\\ &\quad \mathbf{U}_{i+1}(\mathbf{U}_{i+1} + \tilde{\mathbf{Q}}_{i+1}^+)^{-1}\tilde{\mathbf{Q}}_{i+1}^+\tilde{\mathbf{x}}_{i+1}^+],\\ \hat{\mathbf{Q}}_i &= \mathbf{U}_i(\mathbf{U}_i + \tilde{\mathbf{Q}}_{i-1}^-)^{-1}\tilde{\mathbf{Q}}_{i-1}^- + \mathbf{Q}_i^0 +\\ &\quad \mathbf{U}_{i+1}(\mathbf{U}_{i+1} + \tilde{\mathbf{Q}}_{i+1}^+)^{-1}\tilde{\mathbf{Q}}_{i+1}^+.\end{aligned}$$

However, the variables, as applied to the matching problem, are to be restricted by the set of constraints

$$\begin{aligned}\mathcal{X}_i &= \{\mathbf{x}_i : g(\mathbf{x}_{i-1},\mathbf{x}_i) \geq 0, i = 1..n\},\\ g(\mathbf{x}_{i-1},\mathbf{x}_i) &= \mathbf{a}^T(\mathbf{x}_{i-1} - \mathbf{x}_i) - 1.\end{aligned} \qquad (11)$$

In this case, the minimization procedure is fulfilled by the recurrent recalculation of the left-side and right-side Bellman functions (9) in the admissible region (11):

$$\begin{cases}\tilde{J}_i^-(\mathbf{x}_i) = \min\limits_{g(\mathbf{x}_{i-1},\mathbf{x}_i)\geq 0}\left[\psi_i(\mathbf{x}_i) + \gamma_t(\mathbf{x}_{i-1},\mathbf{x}_i) + \tilde{J}_{i-1}^-(\mathbf{x}_{i-1})\right] =\\ \quad = \psi_i(\mathbf{x}_i) + F_i^-(\mathbf{x}_i),\\ \tilde{J}_i^+(\mathbf{x}_i) = \min\limits_{g(\mathbf{x}_{i+1},\mathbf{x}_i)\geq 0}\left[\psi_i(\mathbf{x}_i) + \gamma_t(\mathbf{x}_{i+1},\mathbf{x}_i) + \tilde{J}_{i+1}^+(\mathbf{x}_{i+1})\right] =\\ \quad = \psi_i(\mathbf{x}_i) + F_i^+(\mathbf{x}_i).\end{cases} \qquad (12)$$

Here the functions

$$F_i^-(\mathbf{x}_i) = \min_{g(\mathbf{x}_{i-1},\mathbf{x}_i)\geq 0}\left[\gamma_t(\mathbf{x}_{i-1},\mathbf{x}_i) + \tilde{J}_{i-1}^-(\mathbf{x}_{i-1})\right],$$
$$F_i^+(\mathbf{x}_i) = \min_{g(\mathbf{x}_{i+1},\mathbf{x}_i)\geq 0}\left[\gamma_t(\mathbf{x}_{i+1},\mathbf{x}_i) + \tilde{J}_{i+1}^+(\mathbf{x}_{i+1})\right]$$

are non-quadratic, because, first, the previous Bellman functions $\tilde{J}_{i-1}^-(\mathbf{x}_{i-1})$ and $\tilde{J}_{i+1}^+(\mathbf{x}_{i+1})$ are also non-quadratic and, second, they are formed by minimization under constraints. Therefore, in this case, the whole Bellman functions $\tilde{J}_i^-(\mathbf{x}_i)$ and $\tilde{J}_i^+(\mathbf{x}_i)$ will also be non-quadratic. This fact makes impossible numerical realization of the dynamic programming procedure, since the recurrent recalculation of parameters of the Bellman functions at each step becomes impossible.

Nevertheless, if we suppose that previous Bellman functions are quadratic, then the functions $F_i^-(\mathbf{x}_i)$ and $F_i^+(\mathbf{x}_i)$ can be obtained as result of minimization of respective quadratic functions under linear constraints (11). Both these functions and corresponding Bellman functions are not quadratic only because of the presence of non-equalities in constraints (11).

We consider here one way of overcoming this obstacle, namely, approximate realization of dynamic programming procedure, that consists in substituting functions $F_i^-(\mathbf{x}_i)$ and $F_i^+(\mathbf{x}_i)$ by appropriate quadratic functions

$$F_i'(\mathbf{x}_i) = b_i' + (\mathbf{x}_i - \mathbf{x}_i')^T \mathbf{Q}_i'(\mathbf{x}_i - \mathbf{x}_i') \cong F_i^-(\mathbf{x}_i),$$
$$F_i''(\mathbf{x}_i) = b_i'' + (\mathbf{x}_i - \mathbf{x}_i'')^T \mathbf{Q}_i''(\mathbf{x}_i - \mathbf{x}_i'') \cong F_i^+(\mathbf{x}_i).$$

When such a substitution is done, the next Bellman functions will be quadratic (9), an approximate version of the dynamic programming procedure "forward against forward" will be possible.

The quadratic approximation of the next Bellman functions consists in choosing appropriate parameters of quadratic functions $F_i'(\mathbf{x}_i)$ and $F_i''(\mathbf{x}_i)$, so that the main features of the initial functions $F_i^-(\mathbf{x}_i)$ and $F_i^+(\mathbf{x}_i)$ would be retained. Such features are, first of all, the positions of their minimum points $\arg\min_{\mathbf{x}_i \in R^2} F_i^-(\mathbf{x}_i)$, $\arg\min_{\mathbf{x}_i \in R^2} F_i^+(\mathbf{x}_i)$, and the values of the functions at their minimum points $\min_{\mathbf{x}_i \in R^2} F_i^-(\mathbf{x}_i)$, $\min_{\mathbf{x}_i \in R^2} F_i^+(\mathbf{x}_i)$. It is very important to keep these parameters precisely. Here the position of the minimum point is determined easily by the quadratic programming procedure in R^2. It remains to choose the matrices $\mathbf{Q}'$ and $\mathbf{Q}''$ that define how the quadratic functions $F_i^-(\mathbf{x}_i)$ and $F_i^+(\mathbf{x}_i)$ grow when the argument $\mathbf{x}_i$ deviates from the minimum points $\mathbf{x}_i'$ and $\mathbf{x}_i''$. The proposed approach to the choice of $\mathbf{Q}'$ and $\mathbf{Q}''$ is based on the assumption that the constraints exert the most essential influence on the position of the minimum point $\mathbf{x}_i'$ for $F_i^-(\mathbf{x}_i)$ and, respectively, $\mathbf{x}_i''$ for $F_i^+(\mathbf{x}_i)$, and its quadratic approximation form is of a lesser importance for providing approximation fidelity. We propose to take the matrices of the quadratic functions $\gamma_t(\mathbf{x}_{i-1}, \mathbf{x}_i) + \tilde{J}_{i-1}^-(\mathbf{x}_{i-1})$ and $\gamma_t(\mathbf{x}_{i+1}, \mathbf{x}_i) + \tilde{J}_{i+1}^+(\mathbf{x}_{i+1})$, i.e. matrices defined by formulas $\mathbf{Q}_i' = \tilde{\mathbf{Q}}_{i-1}^-(\tilde{\mathbf{Q}}_{i-1}^- + \mathbf{U}_i)^{-1}\mathbf{U}_i$, $\mathbf{Q}_i'' = \tilde{\mathbf{Q}}_{i+1}^+(\tilde{\mathbf{Q}}_{i+1}^+ + \mathbf{U}_i)^{-1}\mathbf{U}_i$.

5. ALGORITHM OF ELASTIC MATCHING: MINIMIZATION OF PARTIAL OBJECTIVE FUNCTIONS WITH TREE-LIKE SEPARABILITY

It remains to specify how to solve the series of optimization problems (6). Let us consider a single row i_2 of the tree T $(\mathbf{x}_{1,i_2}, \ldots, \mathbf{x}_{n_1,i_2})$ (Fig. 2,b) and associate it with a still more partial objective functions than (6), namely, a pair-wise separable function with chain-like neighborhood-graph R of variables

$$J^{(R,i_2)}(\mathbf{x}_{1,i_2}, \ldots, \mathbf{x}_{n_1,i_2}) = \sum_{i_1=1}^{n_1} \varphi_{i_1,i_2}(\mathbf{x}_{i_1,i_2}) + \sum_{i_1=2}^{n_1} \gamma(\mathbf{x}_{i_1,i_2-1}, \mathbf{x}_{i_1,i_2}) \to \min_{\substack{g(\mathbf{x_i},\mathbf{x_j}) \geq 0, \\ (\mathbf{i},\mathbf{j}) \in R,}} \quad (13)$$

where upper index (R, i_2) is meant to indicate that arguments of this function form the i_2th row in the pixel-neighborhood lattice.

If we give the i_1th variable of function (13) a fixed value $\mathbf{x}_{i_1,i_2}$ and minimize it by other variables, the resulting minimum will be the marginal function of the i_1th variable with respect to the objective function $J^{(R,i_2)}(\mathbf{x}_{1,i_2}, \ldots, \mathbf{x}_{n_1,i_2})$ [3,4]:

$$\hat{J}_{i_1}^{(R,i_2)}(\mathbf{x}_{i_1,i_2}) = \min_{\mathbf{x}_{j_1,i_2},\, j_1 \neq i_1} J^{(R,i_2)}(\mathbf{x}_{1,i_2}, \ldots, \mathbf{x}_{n_1,i_2}). \quad (14)$$

The problem of finding this marginal function can be solved with the help of the procedure of pair-wise quadratic programming described in Section 4.

Then, as it is shown in [4], all the sought-for displacement vectors in the i_1th column of (7) can be found as the solution of the optimization problem

$$J^{(C,i_1)}(\mathbf{x}_{i_1,1}, \ldots, \mathbf{x}_{i_1,n_2}) = \sum_{i_{21}=1}^{n_2} \hat{J}_{i_1,i_2}^{(R)}(\mathbf{x}_{i_1,i_2}) + \sum_{i_2=2}^{n_2} \gamma(\mathbf{x}_{i_1,i_2-1}, \mathbf{x}_{i_1,i_2}) \to \min_{\substack{g(\mathbf{x_i},\mathbf{x_j}) \geq 0, \\ (\mathbf{i},\mathbf{j}) \in T,}} \quad (15)$$

Here the upper index (C, i_1) means that the arguments are the elements of the i_1th column of the pixel grid. Each of these objective functions with chain-like adjacency graph R can be minimized by the pair-wise quadratic programming procedure considered in the previous Section with the only alteration: the respective marginal functions $\hat{J}_{i_1}^{(R,i_2)}(\mathbf{x}_{i_1,i_2})$ obtained earlier are taken instead of the image dependent node functions.

Thus the algorithm of finding the optimal values of the stem-node variables boils down to a combination of pair-wise quadratic programming procedures, each dealing with single image rows, horizontal or vertical ones, considered as signals on the one-dimensional argument axis.

First, such a one-dimensional procedure is applied to the horizontal rows $i_1 = 1, \ldots, n_1$ independently for each $i_2 = 1, \ldots, n_2$. The resulting marginal node functions $\hat{J}_{i_1}^{(R,i_2)}(\mathbf{x}_{i_1,i_2})$ should be stored in the memory. Then, the procedure is applied to the vertical rows $i_2 = 1, \ldots, n_2$ independently for each $i_1 = 1, \ldots, N_1$ with the marginal functions $\hat{J}_{i_1}^{(R,i_2)}(\mathbf{x}_{i_1,i_2})$, obtained at the first step, as the node functions.

6. CONCLUSIONS

Many image processing problems may be represented as those of transformation of the original image into a secondary data field on the same pixel grid. The essence of the problem consists in coordinating the mutually contradictory image-dependent information and smoothness constraints both defined on the pixel neighborhood graph. At the same time, it is just this aspect of the image processing techniques which implies the major part of the amount of processing operations. We have shown that such a generalized problem can be solved by a high-speed procedure of minimizing a separable function defined on a succession of pixel neighborhood trees. The specificity of each particular image processing problem falls only on the way of extracting single units of the local information on the hidden field immediately from the original image, i.e. on the forming of an appropriate image-dependent node function.

REFERENCES

[1] Lades M., Vorbrügen J., Lange J., Malsburg C., Würtz R., Konen W. Distortion invariant object recognition in dynamic link architecture. *IEEE Transactions on Computers, 1993,* Vol. 42,No. 3, pp. 300-311.

[2] V. Mottl, A. Kopylov, A. Blinov, S. Zheltov. Quasi-statistical approach to the problem of stereo image matching. *Proceedings SPIE,* Vol. 2363, 1994, pp. 50-61.

[3] V. Mottl, A. Blinov, A. Kopylov, A. Kostin. Optimization techniques on pixel neighborhood graphs for image processing. In: *Graph-Based Representations in Pattern Recognition* (J.-M. Jolion and W.G. Kropatsch, ed.). Computing, Supplement 12. Springer-Verlag/Wien, 1998, pp. 135-145

[4] I. Muchnik, V. Mottl. *Bellman Functions on Trees for Segmentation, Generalized Smoothing, Matching and Multi-Alignment in Massive Data Sets*. DIMACS Tachnical Report 98-1. Rutgers University, USA, 1998.

[5] R. Fletcher. Practical Methods of Optimizations. John Wiley and Sons Inc., 2nd edition, 1987.

[6] S. German, D. German. Stohastic relaxation. Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. On PAMI*, 1984, 6, pp. 721-741.

[7] A. Kostin, A. Kopylov, V. Mottl, I. Muchnik. Dynamic programming procedures in non-stationary signal analysis. Pattern Recognition and Image Analysis. *Advances in Mathematical Theory and Applications, Vol. 11*, 2001, No. 1, pp. 205-208.