USE OF COLOUR IN GRADIENT-BASED ESTIMATION OF DENSE TWO-DIMENSIONAL MOTION

Janusz Konrad

Institut National de la Recherche Scientifique
INRS-Télécommunications
16 Place du Commerce, Verdun
Québec, Canada, H3E 1H6

Abstract

This paper presents a gradient-based approach to the multi-constraint estimation of dense two-dimensional (2-D) motion. The formulation is based on feature-invariance along motion trajectories and applies motion smoothness constraint to reduce ill-posedness. It permits the use of various image features as the input, for example intensity and colours, or sub-bands of a spectral decomposition. The proposed cost function is minimized using a sequence of quadratic approximations of the matching error and solving the resulting linear system by deterministic relaxation. The proposed algorithm is a generalization of the Horn and Schunck algorithm to the case of vector data. Results of application of the proposed technique to the estimation of 2-D motion from TV images are shown. The obtained motion fields are applied to motion-compensated temporal interpolation resulting in significant but localized improvements.

1. INTRODUCTION

The calculation of dense 2-D motion is usually formulated with respect to a single cue like intensity or luminance. If this cue is locally characterized by little or no detail, motion vectors cannot be reliably computed and are usually derived from their neighbours by smoothing. Since typically images also contain colour information, it is reasonable to expect that the colour cue may contain additional detail. Such detail is usually highly correlated with the intensity, thus is not expected to solve the aperture problem [1], but it may reduce ill-posedness [2] associated with the local lack of intensity detail, and thus improve robustness of motion estimation. In this paper, I address the problem of using additional colour cue to improve robustness of estimation of dense 2-D motion.

Wohn *et al.* [3] and Mitiche *et al.* [4] have proposed multi-constraint methods that augment the single-cue information with signals derived from it (e.g., derivatives, contrast, entropy) or with additional cues (e.g., colour). In both proposals a solution to multi-constraint equations was found using the least-squares approach. Wohn *et al.* computed motion vectors using gradient magnitude, curvature, moments and colour of an image, and subsequently applied smoothing to the resulting motion field. They reported lack of reliability of the method, but Mitiche *et al.* later attributed those findings to an improper choice of constraints based on directional derivatives. They, in turn, used contrast, entropy, average, variance and median of an image to conclude a "reasonable performance of the method even in the presence of moderate noise". Later, a Bayesian approach to the estimation of dense 2-D motion from colour images [5] was proposed. In that approach, the multi-constraint matching modeled by an ensemble of independent, identically distributed (*iid*) Gaussian random variables was combined with a motion smoothness constraint based on a Markov random field (MRF) model. The resulting cost function was solved using the computationally demanding method of *simulated annealing*.

The work presented here aims at incorporating colour information into a more efficient gradient-based framework. The goal is achieved by applying a sequence of quadratic approximations [6] to the Bayesian-formulated cost function proposed in [5]. Each approximation results in a linear system solved using deterministic relaxation. Such an approach is far less demanding computationally than *simulated annealing*, and implemented over a hierarchy of resolutions is expected to be even faster and provide nearly optimal solutions [7]. Moreover, the resulting algorithm is fully parallelizable, thus can be efficiently executed on SIMD (Single Instruction - Multiple Data) architectures. Another goal of this work is to demonstrate practical benefits of using additional colour cue in the estimation of dense 2-D motion.

2. PROBLEM FORMULATION

Let the input consist of K streams of data. The individual streams may correspond to separate visual cues, like intensity and colour, or they may be derived from one signal, for example sub-bands obtained through spectral decomposition. The formulation presented below is general and encompasses both cases as well as other types of input data. Let the individual streams, also called components, be g_k $(k = 1, ..., K)$, and thus let $\boldsymbol{g} = [g_1, g_2, ..., g_K]$ denote the total input signal†. Let $\boldsymbol{x} = [x, y]^T$ and t be

† Terms *signal* and *image* will be used interchangeably in the paper since individual components do not need to correspond to meaningful images, e.g., image entropy or moments.

spatial and temporal positions, respectively, in the input signal. Let the signal $\boldsymbol{g}$ be spatio-temporally sampled with $g_k(\boldsymbol{x}_i, t_j)$ denoting the i-th element (pixel) at j-th time instant of k-th component. Consequently, the dynamics of the input image can be described using motion (displacement) fields $\boldsymbol{d}$ instead of instantaneous velocities. These fields consist of motion trajectories, non-linear in general, which provide a mapping of signal elements between two (or more) time instants. Since it is impossible to calculate motion trajectories over continuum of positions, the sequence of motion fields $\boldsymbol{d}$ is also sampled, however not necessarily on the same structure as the input signal $\boldsymbol{g}$. Thus, $\boldsymbol{d}(\boldsymbol{x}_i, t_j)$ denotes i-th trajectory at j-th time instant. Let the motion field at any time instant contain M vectors.

2.1 MODELS

Estimation of dense 2-D motion is an example of inverse problem often encountered in computer vision. Like other such problems, it is ill-posed, and thus requires a substantial degree of modeling. Firstly, it requires a model of the relationship between input data and estimated values, called here a *structural model.* Secondly, to reduce ill-posedness, it requires a model of the estimated quantity itself, in this case a *motion model.*

As the structural model I will use the standard assumption about invariance of an image feature along motion trajectories. Note, that this assumption is suitable for some features, while it may be unjustified for others. The invariance assumption has been successfully used for luminance and chrominance components of an image [5], based on the observation that colour content in objects is unlikely to move differently than intensity. The assumption holds exactly for the true (unknown) motion field and for an underlying image from which the observed image $\boldsymbol{g}$ is derived if such artifacts as occlusions, illumination variability or object distortions are absent. In a practical system, however, there is always a certain fluctuation of intensity (or colour) along motion trajectories, which is due to optical system imperfections, noise in the camera, aliasing due to sampling, etc.

In this paper I will be concerned with a linear motion model (described below) in which motion trajectories are 3-D vectors. Each such vector has two degrees of freedom because its temporal coordinate is T (temporal sampling period for $\boldsymbol{g}$). Thus, a displacement field $\boldsymbol{d}$ of 2-D vectors, which are spatial projections of the 3-D vectors, is sought. Consider estimation of motion field $\boldsymbol{d}$ at time instant t from two images[†]. Let image at $t_- = t - \alpha \cdot T$ be the preceding image and image at $t_+ = t + (1-\alpha) \cdot T$ be the following one, where $\alpha = \Delta t/T$ and Δt is the temporal distance between the preceding image and the motion field. With the above assumptions, the error resulting from matching two images of component g_k using $\boldsymbol{d}$ can be expressed through the *displaced component difference* (DCD) defined for $i = 1, ..., M$ as

$$\widetilde{r}_{k,i}(\boldsymbol{d}) = \widetilde{g}_k(\boldsymbol{x}_i + (1-\alpha)\boldsymbol{d}(\boldsymbol{x}_i, t), t_+) - \widetilde{g}_k(\boldsymbol{x}_i - \alpha \boldsymbol{d}(\boldsymbol{x}_i, t), t_-), \quad (1)$$

[†] The proposed multi-constraint approach can be easily extended to more than two images.

where $\widetilde{g}_k$ is a spatially interpolated version of g_k needed for locations not on the sampling structure of $\boldsymbol{g}$. Note that the double subscript k, i above signifies the k-th component of the error at spatial position $\boldsymbol{x}_i$. It is difficult to identify properties of $\widetilde{r}$, but since in uniform areas (no artifacts) $\widetilde{r}$ is expected to be largely caused by noise, aliasing, etc., it would seem reasonable to use a random variable, e.g., Gaussian, as a model. Such modeling has been successfully applied in the context of Bayesian formulation [8]. In this work it is assumed that although matching errors $\widetilde{r}_{k,i}$ for the same pixel i are not independent, they are uncorrelated enough to expect additional information from complementary colour cues.

Despite the fact that motion trajectories are rarely linear at a macroscale, they are usually very close to linear at a certain microscale. Thus, it is assumed that for a sufficiently small T motion trajectories are close to linear and can be modeled by vectors. Also, since minimization of the DCD (1) is insufficient for reliable motion estimation due to ill-posedness [2] and aperture effect [1], usually an assumption about motion field smoothness [2] is incorporated into the formulation. The departure from smoothness of a motion vector at location $(\boldsymbol{x}_i, t)$ can be described by the following error term

$$\sum_{j \in \eta_i} (\boldsymbol{d}(\boldsymbol{x}_i, t) - \boldsymbol{d}(\boldsymbol{x}_j, t))^T (\boldsymbol{d}(\boldsymbol{x}_i, t) - \boldsymbol{d}(\boldsymbol{x}_j, t)), \quad (2)$$

where superscript T denotes transposition and η_i is a half-neighbourhood of spatial location $\boldsymbol{x}_i$ containing only eastern and southern neighbours of $\boldsymbol{x}_i$. Note that error (2) is a discrete approximation of a squared magnitude of the gradient [2].

2.2 COST FUNCTION

The goal is to find a motion field which best explains the observed images $\boldsymbol{g}_{t_-}$ and $\boldsymbol{g}_{t_+}$, and simultaneously conforms to *a priori* properties imposed by the motion model. This can be expressed through minimization of a cost function which combines error terms (1) and (2):

$$\min_{\boldsymbol{d}} \sum_{i=1}^{M} \sum_{k=1}^{K} \lambda_{g_k} \cdot \widetilde{r}^2_{k,i}(\boldsymbol{d}) + \lambda_{\boldsymbol{d}} \sum_{j \in \eta_i} (\boldsymbol{d}_i - \boldsymbol{d}_j)^T (\boldsymbol{d}_i - \boldsymbol{d}_j), \quad (3)$$

where λ_{g_k} and $\lambda_{\boldsymbol{d}}$ are weights, and $\boldsymbol{d}_i = \boldsymbol{d}(\boldsymbol{x}_i, t)$ is used to simplify notation.

Minimization (3) is a special case of Bayesian estimation of dense 2-D motion proposed in [5], where *a posteriori* probability of a motion field $\boldsymbol{d}$ given images $\boldsymbol{g}_{t_-}, \boldsymbol{g}_{t_+}$ is maximized. In the Bayesian formulation, each $\widetilde{r}_{k,i}$ is modeled by a random variable while displacement field $\boldsymbol{d}$ is modeled by a random field (random process). It can be shown [5] that for *iid* Gaussian random variables and the first-order MRF model, minimization (3) results.

Minimization (3) can be also viewed as regularization, where the smoothness term reduces ambiguity in matching. Note that (3) extends the discretized form of Horn and Schunck's formulation [2] to the case of vector data.

3. SOLUTION METHOD

Minimization (3) was originally solved in [5] for the case of discrete state space using the method of *simulated annealing*. Under certain conditions this method is able to find the global optimum of any cost function by executing a random search through the space of solutions. Due to randomness of the search, however, stochastic relaxation requires a very high computational effort.

In this paper, I propose to solve (3) using deterministic relaxation. This approach is similar to the one used by Horn and Schunck [2] except for the type of approximation used to linearize the matching error $\widetilde{r}$. Horn and Schunck have used Taylor expansion of $\widetilde{r}$ at $\boldsymbol{d} = 0$, whereas I propose to expand it around an operating point $\boldsymbol{d} = \dot{\boldsymbol{d}}$. Assuming local linearity of the component g_k and disregarding higher than first-order elements, the following approximation holds for the i-th pixel of the k-th component:

$$\widetilde{r}_{k,i}(\boldsymbol{d}) \approx \widetilde{r}_{k,i}(\dot{\boldsymbol{d}}) + \nabla_x^T \widetilde{r}_{k,i}(\dot{\boldsymbol{d}}) \cdot (\boldsymbol{d}_i - \dot{\boldsymbol{d}}_i), \qquad (4)$$

where the spatial gradient is defined as follows:

$$\nabla_x \widetilde{r}_{k,i}(\dot{\boldsymbol{d}}) = \begin{bmatrix} \widetilde{r}^x_{k,i}(\dot{\boldsymbol{d}}) \\ \widetilde{r}^y_{k,i}(\dot{\boldsymbol{d}}) \end{bmatrix} = \begin{bmatrix} \frac{\partial \widetilde{g}(\boldsymbol{x}_i - \alpha \dot{\boldsymbol{d}}(\boldsymbol{x}_i,t),t_-)}{\partial x}\alpha + \frac{\partial \widetilde{g}(\boldsymbol{x}_i + (1-\alpha)\dot{\boldsymbol{d}}(\boldsymbol{x}_i,t),t_+)}{\partial x}(1-\alpha) \\ \frac{\partial \widetilde{g}(\boldsymbol{x}_i - \alpha \dot{\boldsymbol{d}}(\boldsymbol{x}_i,t),t_-)}{\partial y}\alpha + \frac{\partial \widetilde{g}(\boldsymbol{x}_i + (1-\alpha)\dot{\boldsymbol{d}}(\boldsymbol{x}_i,t),t_+)}{\partial y}(1-\alpha) \end{bmatrix}. \qquad (5)$$

Note that approximation (4) simplifies to the one used by Horn and Schunck for K=1, $\dot{\boldsymbol{d}}_i = 0$ and bilinear interpolation used to obtain $\widetilde{g}$.

With this approximation, the cost function $\boldsymbol{E}$ under minimization in (3) becomes quadratic in $\boldsymbol{d}$, and thus the necessary condition for optimality can be established for each vector $\boldsymbol{d}_i$ through:

$$\frac{\partial \boldsymbol{E}_i}{\partial \boldsymbol{d}_i} = 2\sum_{k=1}^{K} \lambda_{g_k} [\widetilde{r}_{k,i}(\dot{\boldsymbol{d}}) + \nabla_x^T \widetilde{r}_{k,i}(\dot{\boldsymbol{d}}) \cdot (\boldsymbol{d}_i - \dot{\boldsymbol{d}}_i)] \nabla_x \widetilde{r}_{k,i}(\dot{\boldsymbol{d}}) + 8\lambda_d [\boldsymbol{d}_i - \overline{\boldsymbol{d}}_i] = 0, \qquad (6)$$

with $\overline{\boldsymbol{d}}_i$ being an average vector calculated over four neighbours around $\boldsymbol{x}_i$, but without $\boldsymbol{x}_i$ itself. The above equation can be rewritten in matrix form:

$$\boldsymbol{D}_i^T \boldsymbol{L} \boldsymbol{R}_i + \boldsymbol{D}_i^T \boldsymbol{L} \boldsymbol{D}_i (\boldsymbol{d}_i - \dot{\boldsymbol{d}}_i) + 4\lambda_d (\boldsymbol{d}_i - \overline{\boldsymbol{d}}_i) = 0, \qquad (7)$$

where $\boldsymbol{R}_i$ and $\boldsymbol{D}_i$ are vectors of DCDs and their gradients at pixel i, respectively, and $\boldsymbol{L}$ is a diagonal matrix of weights λ_{g_k}:

$$\boldsymbol{R}_i = \begin{bmatrix} \widetilde{r}_{1,i}(\dot{\boldsymbol{d}}) \\ \cdot \\ \widetilde{r}_{K,i}(\dot{\boldsymbol{d}}) \end{bmatrix}, \boldsymbol{D}_i = \begin{bmatrix} \nabla_x^T \widetilde{r}_{1,i}(\dot{\boldsymbol{d}}) \\ \cdot \\ \nabla_x^T \widetilde{r}_{K,i}(\dot{\boldsymbol{d}}) \end{bmatrix}, \boldsymbol{L} = \begin{bmatrix} \lambda_{g_1} & \cdot & 0 \\ \cdot & \cdot & \cdot \\ 0 & \cdot & \lambda_{g_K} \end{bmatrix}.$$

Equation (7) can be rearranged to obtain

$$\boldsymbol{d}_i = \dot{\boldsymbol{d}}_i - (4\lambda_d I + \boldsymbol{D}_i^T \boldsymbol{L} \boldsymbol{D}_i)^{-1} [4\lambda_d (\dot{\boldsymbol{d}}_i - \overline{\boldsymbol{d}}_i) + \boldsymbol{D}_i^T \boldsymbol{L} \boldsymbol{R}_i],$$

where I is the identity matrix. This equation must be satisfied by each vector $\boldsymbol{d}_i$ and there are typically several thousands of them. The resulting linear system is sparse, thus instead of simultaneously solving all equations, a relaxation method, like Jacobi or Gauss-Seidel, can be used. Let superscript n denote iteration number. Then, displacement vector $\boldsymbol{d}_i$ can be calculated iteratively as follows

$$\boldsymbol{d}_i^{n+1} = \dot{\boldsymbol{d}}_i - (4\lambda_d I + \boldsymbol{D}_i^T \boldsymbol{L} \boldsymbol{D}_i)^{-1} [4\lambda_d (\dot{\boldsymbol{d}}_i - \overline{\boldsymbol{d}}_i^n) + \boldsymbol{D}_i^T \boldsymbol{L} \boldsymbol{R}_i], \qquad (8)$$

where $\overline{\boldsymbol{d}}_i^n$ is a spatial average obtained in the previous iteration. The algorithm starts from some initial $\dot{\boldsymbol{d}}$ (e.g., zero vector field) and iteratively calculates an estimate of $\boldsymbol{d}$. Upon convergence, the resulting estimate is used as the new $\dot{\boldsymbol{d}}$, and the process is repeated. The adaptation of $\dot{\boldsymbol{d}}$ to the previous solution is very important for the calculation of derivatives (5). With every new $\dot{\boldsymbol{d}}$, which is expected to be better than the previous one, unless the algorithm diverges, a more accurate estimate of derivatives is obtained due to displacement compensation. In the original Horn and Schunck algorithm there is no such compensation when calculating derivatives. For fast motion this may lead to incorrect gradient estimates since derivatives may be calculated over an object in one image and over background in the other. Very significant improvements due to the use of motion-compensated derivatives in the single-constraint case have been reported in [6] (similar method has been proposed by Nagel and Enkelman [9]). A detailed expansion of iterative equation in (8) can be found in the Appendix.

It can be shown that iteration (8) is equivalent to minimization (3) executed using the Gauss-Newton method. However, if it is assumed that $\dot{\boldsymbol{d}} = \overline{\boldsymbol{d}}$, i.e., the operating point is computed at each iteration as an average from the previous estimate, then the following iteration results

$$\boldsymbol{d}_i^{n+1} = \overline{\boldsymbol{d}}_i^n - (4\lambda_d I + \boldsymbol{D}_i^T \boldsymbol{L} \boldsymbol{D}_i)^{-1} \boldsymbol{D}_i^T \boldsymbol{L} \boldsymbol{R}_i, \qquad (9)$$

which is a multi-constraint variant of the Horn-Schunck algorithm with motion-compensated derivatives. Note that in (9) the vectors $\boldsymbol{D}_i$ and $\boldsymbol{R}_i$ are now evaluated at $\dot{\boldsymbol{d}} = \overline{\boldsymbol{d}}$.

4. EXPERIMENTAL RESULTS

The algorithm presented above has been applied to several colour image sequences in the Y-C1-C2 format ($k = 1, 2, 3$). In this format image fields are 2:1 line-interlaced and chrominances C1, C2 are horizontally subsampled by 2 relative to luminance Y. Fig.1.a shows a luminance (Y) window of 224 pixels by 144 lines from one field of sequence "pingpong", while Fig.1.b shows the chrominance C1 for the same window. Both are shown with lines and pixels appropriately interpolated for proper aspect ratio. Note that in Fig.1.a it is difficult to identify the racket boundaries due to similar luminance of the racket and the shirt. In Fig.1.b, however, the racket can be localized more easily because it is of different colour than the shirt. This additional chrominance detail is expected to be helpful when used in the multi-constraint approach.

The multi-constraint algorithm with K=3 has been compared with the algorithm based on luminance only i.e., for K=1. Both have been implemented in a hierarchical framework [10] over 4 levels of resolution with 2×2 subsampling between levels. The image pyramid has been obtained by a cascade of three Gaussian filters, each with 5 independent coefficients and a variance of 2.5. The ratio λ_d/λ_{g_k} over hierarchy of resolutions was controlled similarly to the multiscale MRF model from [7]. To calculate

interpolated values of $\widetilde{g}_k$, bicubic interpolation [11] was used. The tradeoff between matching quality and motion field smoothness was established by the ratio $\lambda_d/\lambda_{g_k} = 6$ for $k=1, 2, 3$, and $\lambda_d/\lambda_{g_1} = 2$ for the luminance-only case.

The proposed method has been tested in the context of motion-compensated interpolation. First, the original sequence has been temporally subsampled by a factor of 2, and motion fields have been estimated from the remaining images. Then, the missing images have been reconstructed using motion-compensated temporal interpolation.

Fig.2 shows examples of motion fields estimated from the sequence "pingpong" using luminance information Y (a) only, and using all three components Y-C1-C2 (b). The Y-based algorithm results in incorrect motion estimates in the center where fast motion of the racket (Fig.1.a) occurs. The Y-C1-C2-based algorithm (b) produces vectors with orientations consistently closer to the true motion. This improvement is confirmed in Fig.3 where reconstructed luminance images using the Y-based (a) and Y-C1-C2-based (b) motion estimation are shown. Note a much improved rendition of the hand with the racket. The improvement is even more pronounced when viewed on a colour TV monitor. It can be examined closer by looking at the reconstruction error (difference between the reconstructed image and the original one) depicted for the luminance component in Figs.4.a and 4.b, where the degree of departure from grey (brighter or darker) is proportional to the error. Note again a significant reduction of the reconstruction error for images recovered using Y-C1-C2-based motion estimation. One should also expect a reduction of the reconstruction error for the chrominance components. This is confirmed in Figs.4.c and 4.d which show the C1 component of the error (the error for C2 component is similar).

Both algorithms have been also applied to the estimation of known motion. Subjectively, the improvement due to the use of colour was localized, however substantial. Also, in terms of the mean-squared error the use of colour was beneficial. For examples with synthetic motion obtained using *simulated annealing*, please consult [5].

The fact that motion estimates derived from simultaneous constraints on Y, C1 and C2 give consistently smaller error for all three image components indicates that they are closer to the true motion field than the ones derived from luminance only. The improvement, however, is limited to the areas where colour detail provides additional information. In this work all three constraints were used homogeneously across the image increasing the computational demand by about three. Since the improvement due to colour is usually localized, it would be beneficial to exploit this additional information only when it is significant, for example by monitoring colour derivatives.

5. SUMMARY AND CONCLUSIONS

A multi-constraint gradient-based approach to the estimation of dense 2-D motion has been proposed in the paper. The resulting algorithm has been shown to be a generalization of the Horn and Schunck method to the vector data case. The proposed method uses motion-compensated image gradient, instead of a non-compensated one, thus increasing estimation robustness at boundaries of fast moving objects. The multi-constraint estimation has been compared with the single-constraint estimation on several colour image sequences in the Y-C1-C2 format. It has consistently provided better motion estimates than the luminance-based algorithm, and has resulted in significantly improved quality of images reconstructed through motion-compensated interpolation. The method, however, is more involved computationally than estimation based on one image component. Currently, means to reduce method's complexity are investigated.

APPENDIX

In order to expand the equation (8), inverse of the 2 x 2 matrix $\boldsymbol{A} = 4\lambda_d I + \boldsymbol{D}_i^T \boldsymbol{L} \boldsymbol{D}_i$ must be found. To simplify derivations in this appendix, dependence of $\widetilde{r}$, $\widetilde{r}^x$, $\widetilde{r}^y$ and $\nabla_x \widetilde{r}$ on i and $\dot{\boldsymbol{d}}$ will be temporarily omitted, as well as dependence of $\boldsymbol{d}$, $\dot{\boldsymbol{d}}$, $\overline{\boldsymbol{d}}$ on i. The summations extend from 1 to K unless otherwise indicated.

By carrying out matrix multiplications it can be shown that:

$$\boldsymbol{A} = \begin{bmatrix} 4\lambda_d + \sum_k \lambda_{g_k} (\widetilde{r}_k^x)^2 & \sum_k \lambda_{g_k} \widetilde{r}_k^x \widetilde{r}_k^y \\ \sum_k \lambda_{g_k} \widetilde{r}_k^x \widetilde{r}_k^y & 4\lambda_d + \sum_k \lambda_{g_k} (\widetilde{r}_k^y)^2 \end{bmatrix}.$$

Thus, the inverse of $\boldsymbol{A}$ is

$$\boldsymbol{A}^{-1} = |\boldsymbol{A}|^{-1} \begin{bmatrix} 4\lambda_d + \sum_k \lambda_{g_k} (\widetilde{r}_k^y)^2 & -\sum_k \lambda_{g_k} \widetilde{r}_k^x \widetilde{r}_k^y \\ -\sum_k \lambda_{g_k} \widetilde{r}_k^x \widetilde{r}_k^y & 4\lambda_d + \sum_k \lambda_{g_k} (\widetilde{r}_k^x)^2 \end{bmatrix},$$

with $|\boldsymbol{A}|$ being the determinant defined as follows

$$|\boldsymbol{A}| = \sum_k \lambda_{g_k} \left[4\lambda_d \left((\widetilde{r}_k^x)^2 + (\widetilde{r}_k^y)^2 \right) + \sum_{l<k} (\widetilde{r}_k^x \widetilde{r}_l^y - \widetilde{r}_k^y \widetilde{r}_l^x)^2 \right] + 16\lambda_d^2. \tag{A.1}$$

This form of $|\boldsymbol{A}|$ has been obtained by decomposing the original summation over rectangle $(k=1, ..., K; l=1, ..., K)$ into summations over lower and upper triangles (summation along the main diagonal gives zero). It can be easily demonstrated that matrix $\boldsymbol{B} = \boldsymbol{D}_i^T \boldsymbol{L} \boldsymbol{R}_i$ is defined as follows:

$$\boldsymbol{B} = 4\lambda_d (\dot{\boldsymbol{d}} - \overline{\boldsymbol{d}}) + \sum_k \lambda_{g_k} \widetilde{r}_k \nabla_x \widetilde{r}_k.$$

It remains to carry out the multiplication of $\boldsymbol{A}^{-1}$ and $\boldsymbol{B}$. After somewhat tedious calculations one obtains the following iterative update equation:

$$\boldsymbol{d}^{n+1} = \overline{\boldsymbol{d}}^n - |\boldsymbol{A}|^{-1} \cdot \begin{bmatrix} \sum_k \lambda_{g_k} \left(\widetilde{r}_k + \nabla_x^T \widetilde{r}_k (\overline{\boldsymbol{d}}^n - \dot{\boldsymbol{d}}) \right) \left(4\lambda_d \widetilde{r}_k^x + \sum_l \lambda_{g_l} \widetilde{r}_l^y c_{k,l} \right) \\ \sum_k \lambda_{g_k} \left(\widetilde{r}_k + \nabla_x^T \widetilde{r}_k (\overline{\boldsymbol{d}}^n - \dot{\boldsymbol{d}}) \right) \left(4\lambda_d \widetilde{r}_k^y - \sum_l \lambda_{g_l} \widetilde{r}_l^x c_{k,l} \right) \end{bmatrix}, \tag{A.2}$$

where $c_{k,l} = \widetilde{r}_k^x \widetilde{r}_l^y - \widetilde{r}_k^y \widetilde{r}_l^x$. Recall that iterative equation (A.2) is valid for the i-th vector and that values of $\widetilde{r}$, $\widetilde{r}^x$, $\widetilde{r}^y$ and $\nabla_x \widetilde{r}$ are computed for $\dot{\boldsymbol{d}}$. Modification to obtain equation (9) is obvious.

For the case of scalar input data, i.e., $K = 1$, the summations with respect to l in (A.1) and (A.2) give zero ($c_{k,k} = 0$), thus resulting in an iterative equation of the modified Horn-Schunck algorithm presented in [6].

REFERENCES

[1] E. C. Hildreth, "Computations underlying the measurement of visual motion," *Artificial Intell.*, vol. 23, pp. 309–354, 1984.

[2] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intell.*, vol. 17, pp. 185–203, 1981.

[3] K. Wohn, L. S. Davis, and P. Thrift, "Motion estimation based on multiple local constraints and nonlinear smoothing," *Pattern Recognition*, vol. 16, no. 6, pp. 563–570, 1983.

[4] A. Mitiche, Y. F. Wang, and J. K. Aggarwal, "Experiments in computing optical flow with the gradient-based, multiconstraint method," *Pattern Recognition*, vol. 20, no. 2, pp. 173–179, 1987.

[5] J. Konrad and E. Dubois, "Use of colour information in Bayesian estimation of 2-D motion," in *Proc. IEEE Int. Conf. on Acoust. Speech Signal Process.*, pp. 2205–2208, Apr. 1990.

[6] J. Konrad and E. Dubois, "Comparison of stochastic and deterministic solution methods in Bayesian estimation of 2D motion," *Image & Vision Computing*, vol. 9, pp. 215–228, Aug. 1991.

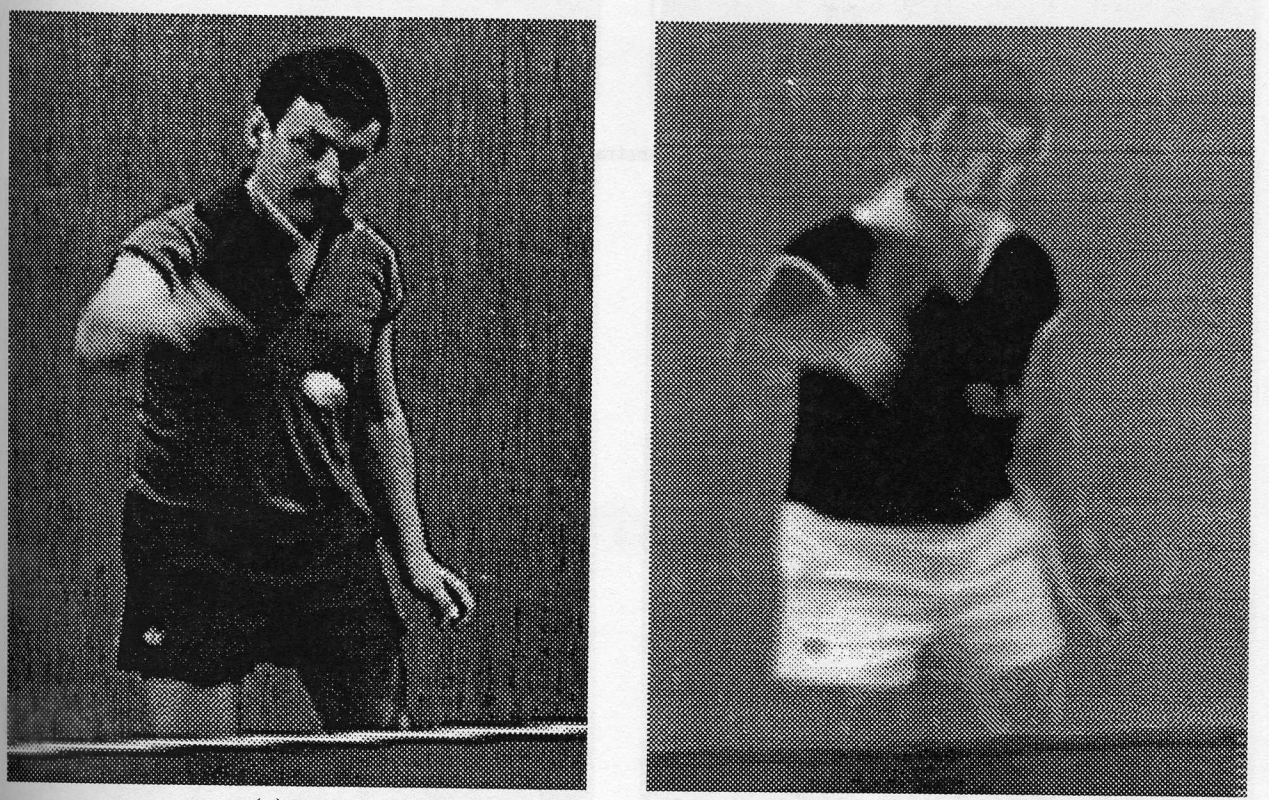
[7] F. Heitz, P. Perez, and P. Bouthemy, "Parallel visual motion analysis using multiscale Markov Random Fields," in *Proc. IEEE Workshop Motion*, Oct. 1991.

[8] J. Konrad and E. Dubois, "Estimation of image motion fields: Bayesian formulation and stochastic solution," in *Proc. IEEE Int. Conf. on Acoust. Speech Signal Process.*, pp. 1072–1075, Apr. 1988.

[9] H. H. Nagel and W. Enkelmann, "An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, pp. 565–593, Sept. 1986.

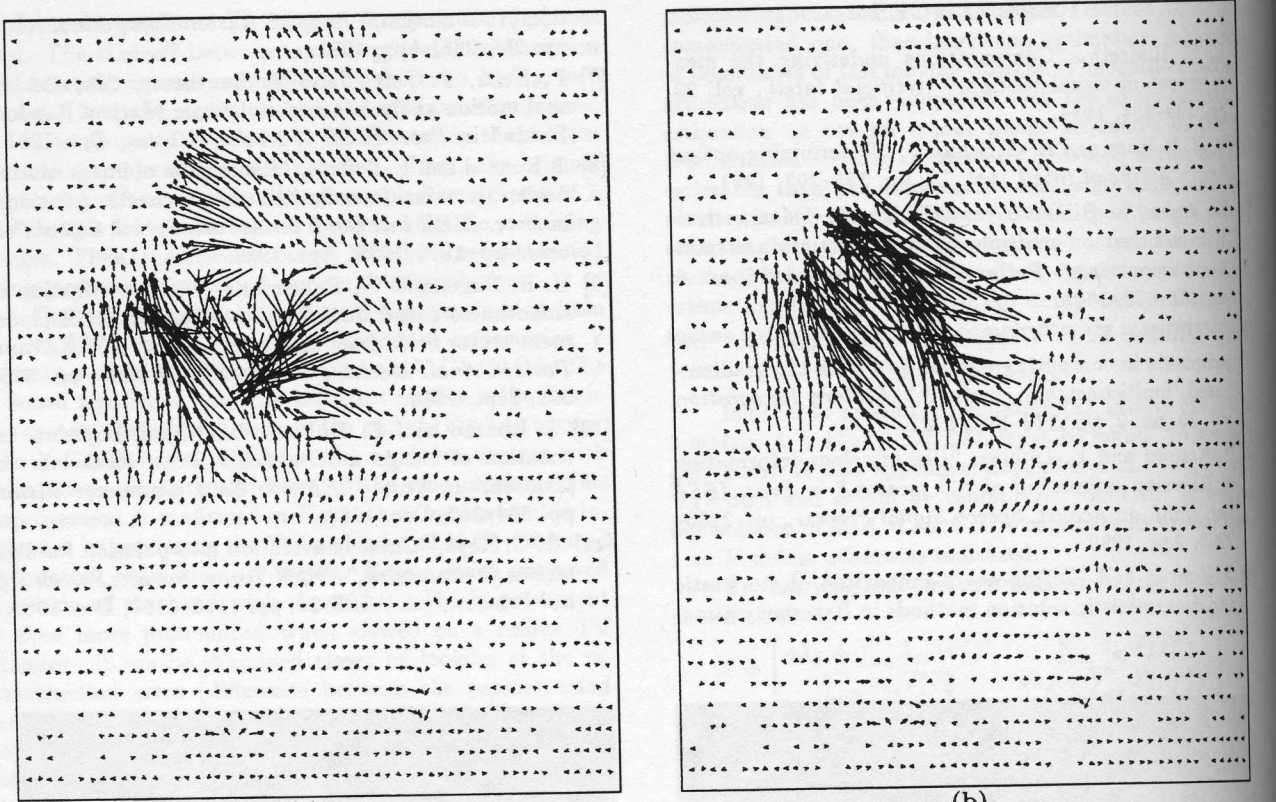
[10] J. Konrad and E. Dubois, "Multigrid Bayesian estimation of image motion fields using stochastic relaxation," in *Proc. IEEE Int. Conf. Computer Vision*, pp. 354–362, Dec. 1988.

[11] R. G. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust. Speech Signal Process.*, vol. ASSP-29, pp. 1153–1160, Dec. 1981.



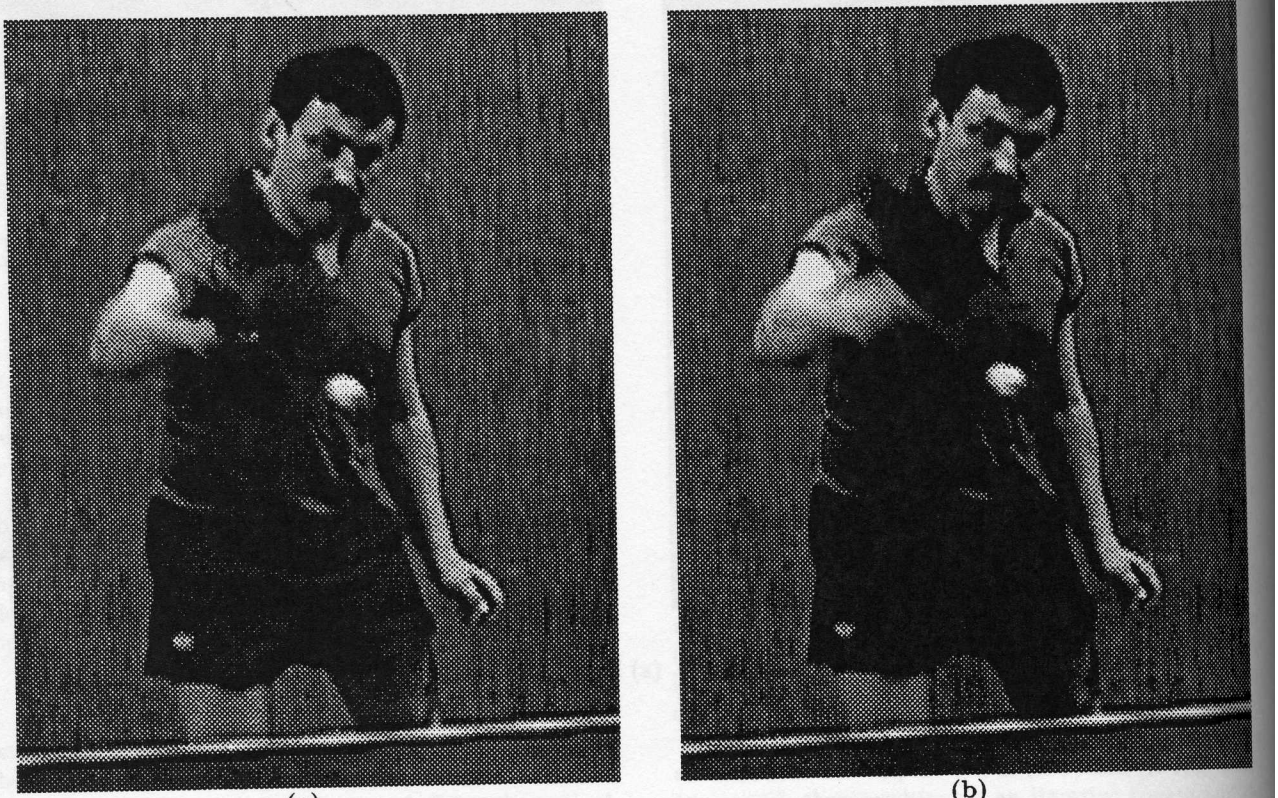
(a) (b)

Fig. 1 Test image "pingpong": (a) luminance Y; (b) chrominance C1.



(a) (b)

Fig. 2 Motion estimates obtained with Y constraint only (a); and with Y-C1-C2 constraints (b).



(a) (b)

Fig. 3 Reconstructed luminance images using Y-based (a); and Y-C1-C2-based estimation (b).

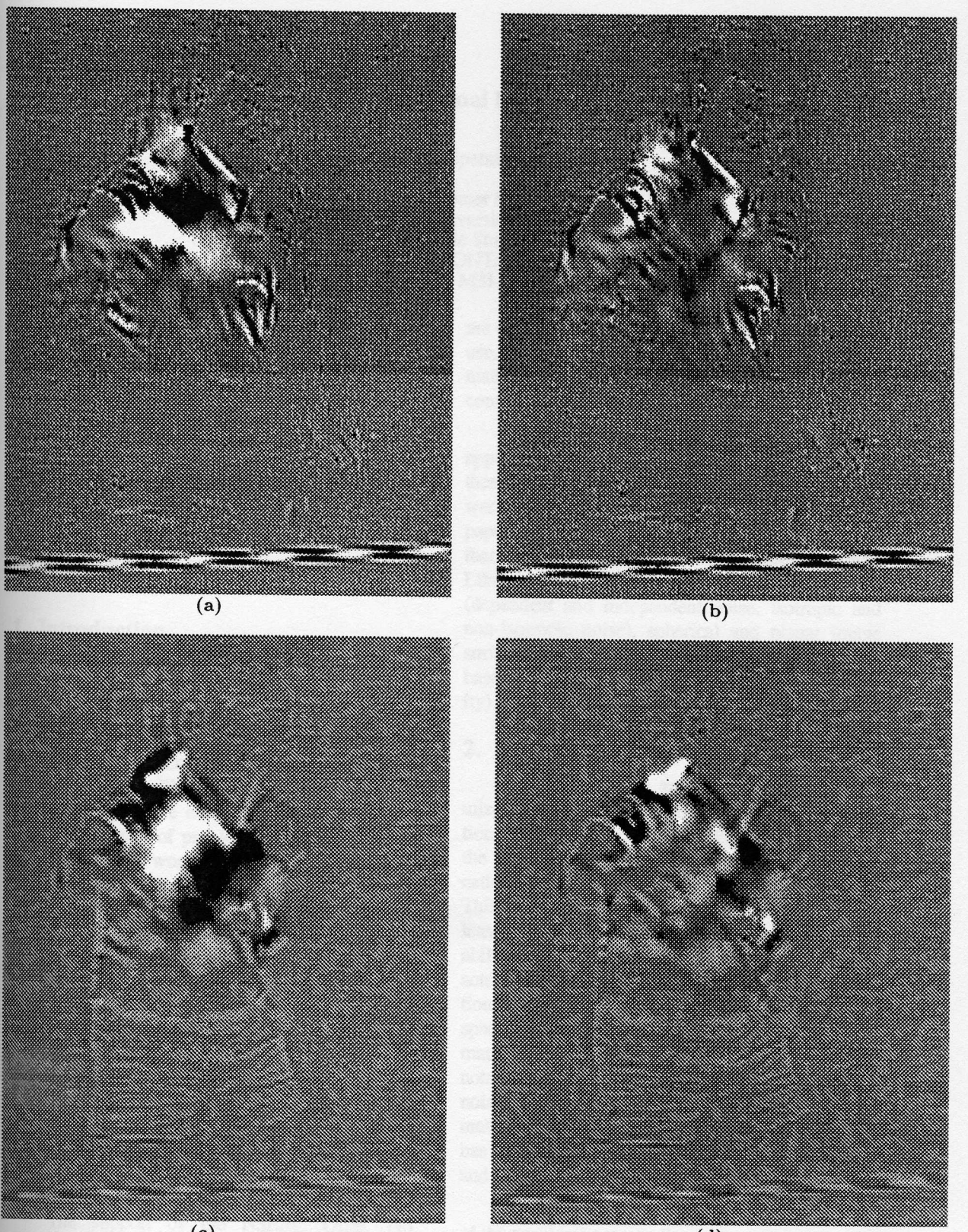


Fig. 4 Luminance Y (a,b) and chrominance C1 (c,d) reconstruction error using Y-based (a,c); and Y-C1-C2-based estimation (b,d).