Periodicity-Based Computation of Optical Flow

Georgii Khachaturov, Silvia Beatriz González Brambila, and Jesús Isidro González Trejo

Departamento de Sistemas, Universidad Autónoma Metropolitana (Azcapotzalco)
Mexico

xgeorge@correo.azc.uam.mx

Abstract. The standard Brightness Constancy Equation states spatiotemporal shift invariance of the input data along a local velocity of optical flow. In its turn, the shift invariance leads to a periodic function of a real argument. This allows application of a known test for periodicity to computation of optical flow at random locations. The approach is valid also for higher dimensions: for example, it applies to a sequence of 3D tomography images. The proposed method has a reasonably high accuracy for continuous flow and is noise tolerant. Special attention is paid to weak signal input. It is shown that a drastic reduction in the signal strength worsens the accuracy of estimates insignificantly. For a possible application to tomography, this would lead to an unprecedented diminution of harmful radiation exposure.

Keywords. Optical flow, periodicity-based processing, preventive tomography, night vision.

1 Introduction

It turns out to be that for an image sequence, the pair {*Velocity of Optical Flow* (VOF); *a patch of the input dataset*} naturally defines a periodic function of a real argument. Using this observation, a novel local detector of optical flow is proposed in this paper. The detector treats a family of functions indexed by velocities of a range; any function of the family is subjected to a test for periodicity, and the estimate of VOF is set to the index of the 'most periodic' function in the family.

We show that a particular test for periodicity chosen in our implementation of this scheme provides the detector with noise tolerance and a reasonably high accuracy while treating datasets of a continuous flow.

Then we show why the noise tolerance is relevant for applications. In particular, it turns out to be that determining optical flow for a class of 'non-hard' datasets can be done practically without a loss of performance by images of a much lower quality than usual. As a practical consequence of this property, the exposure while shooting images for the purpose of automatic determining optical flow can be reduced by several hundred times compared to the images needed for a visual analysis by an expert. This opportunity opens in particular the way for a new technology in medical image processing: a preventive tomography.

Noise tolerance is studied here only with respect to Poisson noise because it overrides all other kinds of noise for weak signal images which are of a particular importance for applications.

1.1 Prior Approaches

The problem of determining optical flow has been attracting numerous researchers since the early 1980s. Interest to this field is stimulated by an increasing demand from applications that deal with temporal sequences of images: video compressing, security systems, medicine, traffic control, and robotics, among others.

Optical flow is a vector field in the image frame. Any vector of this field signifies velocity. In turn, the term *velocity* allows two interpretations: (i) the velocity of a projection of a real point; (ii) the velocity of image motion regardless of how the image content represents the real objects.

These two meanings do not always coincide (Marr and Ullman [21], Horn [12]). To realize that, imagine a thin black ring on a white background and assume that the ring diameter grows from shot to shot in an input sequence of shots; meanwhile the ring center does not move. Under these assumptions it is impossible to answer such questions as "Has the ring a non-zero angular

velocity?" or "Is the ring a part of the background?", or "Is it getting closer to the observer?" On the other hand, the growth of the ring diameter can be determined. So, only meaning (ii) is valid for this example.

However, the velocity in meaning (i) can be determined provided that the ring and background are painted with substantially non-uniform textures.

Following the taxonomy of Fleet and Jepson [9], *full velocity* or just *velocity* is a vector in the above meaning (i); *component velocity* or *normal velocity*, in meaning (ii). The corresponding fields in Horn's taxonomy [12] are *motion field* and *apparent motion*.

Considering a method, it is important which of the two interpretations is assumed because if determining optical flow is not feasible by local processing, it might become feasible in a wider area and under additional assumptions. This ambiguity is known as *the aperture problem* [21]. For instance, substitute the ring mentioned above with a rectangle; then velocities can be computed locally only for the corners of the rectangle; assuming the rectangle to be stiff, the corner estimates can be extrapolated to the whole perimeter of the rectangle.

In this respect, the output of our detector represents an estimate of velocity, but in a degenerated case it automatically becomes an estimate of component velocity.

Beginning with the classic works dedicated to optical flow algorithms (Horn and Schunck [13], Lucas and Kanade [20]), the starting point for most approaches is the standard Brightness Constancy Equation (BCE)

$$I(x, y, t) = I(x+u, y+v, t+1) \qquad (1)$$

where $I(x, y, t)$ is the intensity of a pixel (x, y) at time t and the flow is $\{u(x, y, t) , v(x, y, t)\}$. Below, the notation $\alpha = \{u(x, y, t) , v(x, y, t)\}$ is applied for brevity.

BCE is violated when discontinuities, occlusions, shadows, etc. hold. The modern methods compete mainly in trying to improve performance for such special cases. As Baker *et al.* indicate in their survey ([2], Eq. (1)), most existing algorithms pose the problem of determining optical flow as the optimization of a global energy function $E_{Global} = E_{Data} + \lambda E_{Prior}$, where the *Data Term* measures how consistent the optical flow is with input images, and the *Prior Term* favors certain field over others. In practical terms, the data term is responsible for situations when BCE holds, whereas the prior term, for special cases.

The novelty of our approach concerns the data term only. It should be emphasized from the beginning, because the datasets of the commonly accepted modern benchmarking platform (Middlebury open evaluation database, Baker *et al.* [2]) have a strong bias to special cases.

Following [13] and [20], the differential methods rely upon Optical Flow Constraint Equation (OFCE) derived from BCE:

$$u \,\partial I/\partial x + v \,\partial I/\partial y + \partial I/\partial t = 0 \qquad (2)$$

Since Eq. (1) leads to an ill-posed problem, it must be regularized by the prior term. From Eq. (2), Horn and Schunck [13] derive a system of differential equations in partial derivatives that can be solved by processing images of a sequence inside the full frame; so their method is global and its solution is constructed by an iterative relaxation algorithm. In contrast, the method of Lucas and Kanade [20] is local, it can be applied inside a patch of the frame; numerically, it can be regarded as an instance of the least squares method. Bruhn *et al.* [7] present a combined local-global method. Nagel [22] and Uras *et al.* [33] deal in their approaches with partial derivatives not only of the first order, but of the second as well. Variational methods (for example, those of Zimmer *et al.* [36] and Werlberger *et al.* [34]) are derived from the Horn and Schunck [13] approach by modifications of the original data and prior terms. Direct matching of patches for different images of a sequence is an instance of non-differential approaches. It does not develop the data term far beyond BCE. The technique by Anandan [1] is of this kind; it makes use of a Laplacian pyramid and a coarse-to-fine matching strategy. This technique weakly employs priors in the form of a smoothness constraint on the velocity estimates; the solution is based on Gauss-Seidel iterations. A two-stage matching technique is presented by Singh [30].

One more class of the methods is based on spectral techniques that make use of Fourier domain to solve the original spatiotemporal problem. This class includes also the energy-based methods like that of Heeger [11] and the cross correlation and the phase correlation methods (Reddy and Chatterji [26]). The phase-based technique of Fleet and Jepson [9] makes use of the phase information for measurements of component velocity; it deals with a family of spatiotemporal velocity-tuned filters that permits multiple estimates within a single neighborhood; the resolution of individual estimates is reasonably high and so, if the neighborhood represents a patch of a real textured surface, these multiple component velocity estimates lead to a single estimate of the patch velocity.

Brox and Malik [6] introduce rich descriptors into variational technique to cope with a large displacement flow; Xu *et al.* [35] introduce segmentation for a better accuracy. Goldluecke and Cremers [10] show how multi-labeling problem can be applied for determining optical flow. Methods like those of Jepson and Black [16], Jojic and Frey [17], Sudderth *et al.* [31], etc. deal with layered optical flow formed as a sum of several flows. Usually, they work as extraction of a parametric motion for each layer. Genetic algorithms also can be used for computation of flow (Tagliasacchi [32]).

The first comparative study of various methods was performed by Barron *et al.* [4]. It was a long-time reference point for subsequent approaches. Nowadays the Middlebury open evaluation database [2] plays the role of the main benchmarking tool. It provides a considerable progress of objectiveness for methods' evaluation since the authors do not have access to the ground truth of the evaluation datasets. It is also important that the evaluation system automatically generates a series of publicly available reports which contain a comparison of performance of all methods evaluated so far. Any new method submitted for evaluation automatically changes the content of the reports.

The standard Middlebury datasets processed by our method were submitted to the evaluation system. Using the reports of the system, we present in Section 4 a comparison of our method with those that were submitted to the system at the moment of writing.

Unfortunately, at the time of writing there was no system available for the optical flow community that would give an independent evaluation of noise tolerance. This is why the evaluation of noise tolerance of our periodicity-based approach is provided by a different technique which makes use of the datasets provided with ground truth.

Further references to existing methods can be found in the survey in [2].

1.2 Noise Tolerance

Not all methods for determining optical flow are noise tolerant. Indeed, the derivatives in Eq. (2) cannot be computed accurately under noise and therefore the differential methods are incapable of coping with a strong noise.

Potentially, those methods that do not employ derivatives of the input data may be tolerant to noise. The method of Fleet and Jepson [9] is noise tolerant. Nevertheless, we are unaware of any comparative evaluation of noise tolerance of different methods. Those works, including [9], that present experiments with noisy input, do not convey sufficient information for benchmarking noise tolerance.

Note, however, that noise tolerance is important for some applications. Night vision systems, for instance, deal with weak signal that leads inevitably to a high level of Poisson noise. For a sufficiently weak signal, all noise-forming factors become negligible compared to the probabilistic nature of pixel values. It is well known that Poisson distribution represents an adequate statistical model for pixel values regarded as random variables, whereas for the Poisson distribution, a linear reduction in the noise level leads to a quadratic growth of the signal magnitude.

Using this standard fact, we study accuracy of our method for different levels of noise and prove that a dramatic reduction in the signal magnitude can be provided almost without a loss of accuracy. This fact creates new opportunities for application of optical flow, for instance, in the X-ray medical radiography.

1.3. Main Idea

The periodicity-based approach is implemented as a local early vision detector for determining velocity at random locations of the frame. A master-level algorithm calls the detector and passes a location as an argument. The detector response can be negative or positive; a positive response includes a velocity estimate and its covariance matrix. A degenerated covariance matrix means that the estimate is a component velocity.

The following three paragraphs clarify the key details of this scheme.

Velocity is related to shift invariance. The idea of the detector is based on the fact that BCE, Eq. (1), states local shift invariance of the input dataset along the vector $\{u(x,y,t),\ v(x,y,t),\ 1\}$ denoted below as α*. In particular, shift invariance holds in any 2D spatiotemporal profile X_α parallel to α*, Fig. 1. The portion of data that X_α cuts from an input set formed by four shots is depicted in Fig. 1 as bold parallel horizontal lines.
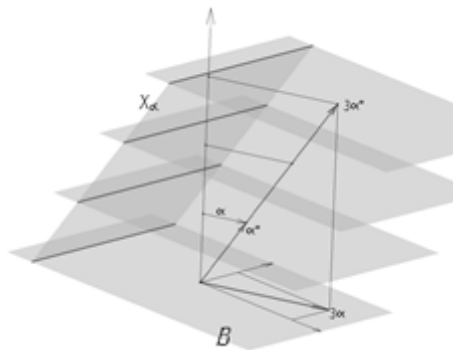

Fig. 1. How search for velocity can be reduced to 3FM

Shift invariance leads to periodicity. Consider a reduction of any input image to the corresponding line of the just mentioned horizontal lines. Due to shift invariance, the functions defined this way represent several identical copies of the same function. Being concatenated, these copies yield a periodic function defined on a long line and the period is equal to the common length of short lines.

A known test for periodicity can be adapted to estimate velocity. If **α** is an arbitrary vector, the concatenated function can be constructed formally in the same way as above. However, this function is not periodic in general.

Our detector compares the outputs of a test for periodicity for different values of **α** and determines the estimate $\tilde{\alpha}$ as such **α** that corresponds to the 'most periodic' function. We make use of an existing test for periodicity —the Three Frequencies Method (3FM) of Khachaturov [18, 19]—that yields quite accurate estimates of period and is noise tolerant. An obstacle arises for direct application of 3FM to each profile X_α: the amount of information provided by a few horizontal layers (which represent shots) is insufficient for accurate estimation. Nevertheless, this drawback is compensated by the fact that many such profiles can be constructed in a small neighborhood. Information from all profiles is accumulated and then used in the 3FM.

The remainder of the paper proceeds as follows: Section 2 describes the 3FM and presents its adaptation to velocity computation as well as a RANSAC-based improvement of the approach to cope with the case when the final estimate must be chosen from several candidates; it presents also an estimate of the dynamic error of the method. Section 3 describes the noise generation. Section 4 presents experiments.

2 Determining Velocity by the 3FM
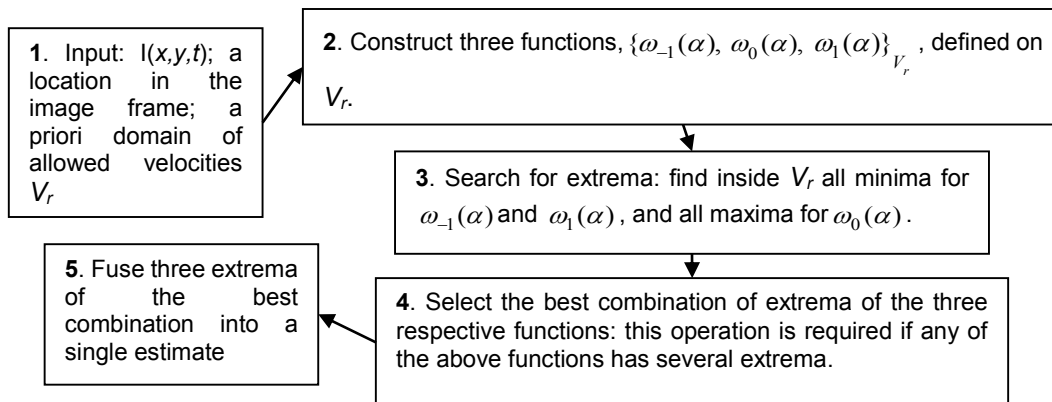
2.1 Periodicity Test of the 3FM and Its Adaptation

The test for periodicity of the 3FM is based on the following

Theorem (Khachaturov [18]**).** Let $f(x)$ be a real function of period P such that for any $x \in \mathrm{R}^1$ it has expansion in Fourier series, T be a natural number, δ be a small real number, and ω be a functional defined as

$$\omega(f,\delta,T,P) = \tfrac{1}{TP}\int_0^{TP} f(x)e^{-i(1+\delta)x\frac{2\pi}{P}}dx, \quad (3)$$

then:

(a) the value of $|\omega(f,0,T,P)|$ does not depend on T;

1. Input: I(x,y,t); a location in the image frame; a priori domain of allowed velocities V_r

2. Construct three functions, $\{\omega_{-1}(\alpha),\ \omega_0(\alpha),\ \omega_1(\alpha)\}_{V_r}$, defined on V_r.

3. Search for extrema: find inside V_r all minima for $\omega_{-1}(\alpha)$ and $\omega_1(\alpha)$, and all maxima for $\omega_0(\alpha)$.

5. Fuse three extrema of the best combination into a single estimate

4. Select the best combination of extrema of the three respective functions: this operation is required if any of the above functions has several extrema.

Fig. 2. Block diagram for periodicity-based velocity estimation

(b) for any $\delta \neq 0$, it is true that $\omega(f,\delta,T,P) \to 0$ as $T \to \infty$;

(c) $\omega(f,1/T,T,P) = \omega(f,-1/T,T,P) = 0$.

The proof can be found in [19]. The notation used in this theorem corresponds to the content of Fig. 1 as follows: T stands for the number of available frames (horizontal layers in Fig. 1); P is the length of time-constant segments of X_α; f represents concatenation of all functions defined on the time-constant segments, so x belongs to [0, TP]. The equation $x=(t-1)P+l$, where $l \in [0,P]$ is real and $t \in [1,T]$ is natural, establishes a one-to-one correspondence $x \leftrightarrow \{t, l\}$ between the points of the long line and the time-constant segments.

Given X_α, this description determines completely the computation of $f(x)$ at any x.

The theorem leads to the following rule: given a family $\{\alpha\}$ of velocity candidates, the velocity estimate is constructed as such $\tilde{\alpha} \in \{\alpha\}$ that the following (i) and (ii) hold simultaneously:

(i) $|\omega_0(\tilde{\alpha})|$ is a local maximum with respect to small variation of α;

(ii) both $\omega_{-1}(\tilde{\alpha})$ and $\omega_1(\tilde{\alpha})$ are zeros.

In practice, instead of (ii), we use the following modification:

(ii*) both $|\omega_{-1}(\tilde{\alpha})|$ and $|\omega_1(\tilde{\alpha})|$ are minima close to zero.

The main steps of this estimation scheme are summarized in the block diagram of Fig. 2.

Some instances of real application of our local detector of velocity are presented in Fig. 3. Each instance is visualized as three plots related to processing respective components of $\{\omega_{-1}(\alpha),\ \omega_0(\alpha),\ \omega_1(\alpha)\}$. The velocity candidates α are represented by nodes shown on the 'floor' of each plot. At any node α, a white post with a black head is depicted; its height represents a normalized value of a corresponding $\omega_i(\alpha)$, where any $i = -1, 0, 1$ represents the plot number. The three functions interpolated from these data are shown both by the surface and by intensity variation of the floor color. Note that for the central plot, a higher intensity stands for a larger value, whereas for the other two, for a smaller one. The three small bright squares depict in triplicate the velocity estimate. These typical examples include an instance of a 'good' case (the top), an instance of a badly conditioned Hessian (the middle), and an instance of several local extrema (the bottom).

Some specific details of this technique are briefly commented upon in the remainder of this section.

– Accumulation of information from a set of profiles (this issue was mentioned in the end of Section 1.3). Technically, it is implemented as a simple sum of square norms of the corresponding individual functionals computed for each profile, like X_α in Fig. 1, of a set. In turn, the set of profiles can be constructed in different ways. In our implementation, given a
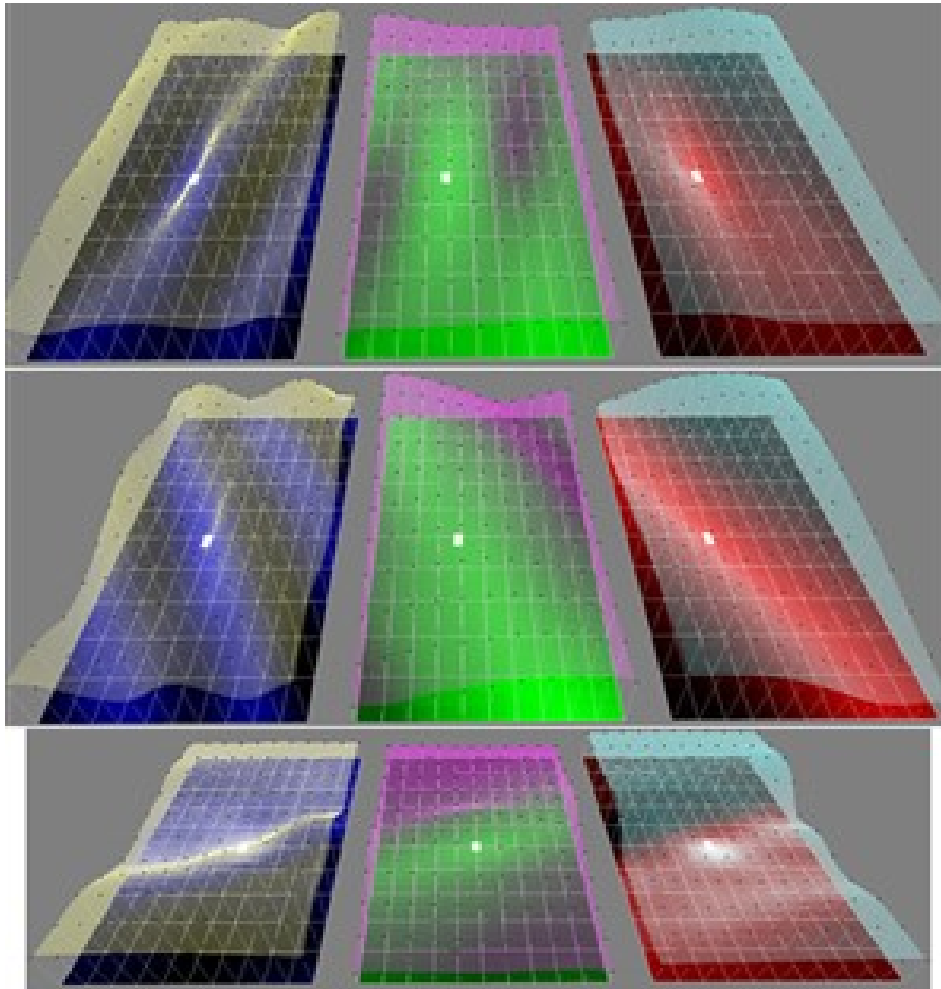
Fig. 3. Visualization of three instances of real application of the detector

location in the frame, a small square domain *B* is constructed around it. Then, a set of lines parallel to a side of *B* is constructed with a fixed step between two adjacent lines. Each line determines a 2D-profile parallel to vector **α***. Two sides of *B* with orthogonal directions are treated in the same way, which duplicates the final number of involved profiles. The size of *B* must be properly balanced: a large size increases both the dynamic error and computational costs and should be avoided at discontinuities, whereas a small size makes the content of patches too uniform.

– Treating color images. The above scheme is applied to every color component and the output functionals are constructed as their sum.

– The case of higher dimensions. Extending our scheme to higher dimensions (e.g., to the temporal sequences of tomography images) is straightforward: the square *B* above is simply substituted with a cube, whereas all the rest remains virtually unchanged.

– Weakening the 3FM. Employing the three independent conditions represented by (i)-(ii*) is important for accuracy of estimation and

suppressing false detection. However, the number of conditions can be reduced in practice: for a vector to be a velocity estimate, it suffices to specify that any two of the three conditions would hold. This variant of the 3FM is important for low textured fragments: it leads to diminution of negative responses (misses) of the detector.

– Fusing extrema of the best combination. The arguments of two or three extrema determined by (i)–(ii*) are fused by the method of least squares (LSM) [25] to obtain the final estimate of velocity. LSM deals with the weights formed by matrices inverse to covariance matrices of the arguments of respective extrema. Any such matrix coincides up to a coefficient with the Hessian that can be easily computed at the corresponding extremum.

– Multi-scaling and cross-scale estimates. Let square B above be supplied with index s, $B=B_s$, that represents the size of the square. Below, the term *scale* stands for s. A cross-scale estimate can be constructed as such one for which any of conditions (i)–(ii*) holds for its own scale. Since conditions (i)–(ii*) must hold theoretically for any scale, the cross-scale and the single-scale estimates have the same theoretical basis.

2.2 The Case of Several Extrema

Given a location in the image frame and a square of allowed velocities $V_r =[-r, r] \times [-r, r]$, where $r>0$, an attempt to apply the method of Section 2.1 can lead to one of the following cases: (a) there is a unique estimate in V_r that satisfies the above conditions (i)–(ii*); (b) there are several candidates to represent the estimate.

In experiments on standard datasets, the first option ('good case') holds for about 80-90% of locations. But difficulties arise when several extrema occur: any local strategy has a non-zero probability of a wrong choice of extremum. Moreover, wrong decisions have usually a high correlation for close nodes of a grid. While processing the famous Yosemite sequence by a straightforward version of the above method, these issues lead to a small number of fluctuations, say, for 0.25-0.5% of all nodes. But these fluctuations

are strong and deteriorate the overall end-point error significantly.

In the remainder of this section, we shall briefly present some ideas to improve performance of the method. We combine the local extrema of several close nodes and several scales (see the end of the previous section). The goal is to cope better with case (b).

In the enhanced technique, first of all, *the set of candidates is extended with respect to those specified by* (i)–(ii*) of Section 2.1. The two ways mentioned in the end of the section are employed for that: the requirement of the three independent conditions is weakened to two, and the cross-scale estimates are also included into the set of candidates. To clarify this heuristic, note that while violations of BCE occur, they deteriorate the extrema exploited in the 3FM, but the deterioration is not uniform for different scales, so there is a chance to rescue a relevant extremum by variation of scale.

The subsequent task is to choose a single estimate among the extended set of candidates.

Some local measures for sorting candidates can be used for this task, but none of them excludes the possibility of a wrong decision. For instance, the extrema of a functional can be ranked by their values and then the rank can be used for choice. Our experiments show that a strategy based on the main idea of RANSAC [8] overrides any local measure strategy. In the rest of the section we comment on this issue.

To determine the flow in the dense domain, we apply the detector not for all pixels, but for nodes of a sparse regular grid. Then these results are interpolated for the whole domain. The step between the nodes of the grid is set to the smallest scale. To implement RANSAC-based strategy, we consider the 3×3-window of the grid around any current node, then a function $S(c)$ is introduced as the number of nodes of the window that have a candidate close to the candidate c. The estimate is constructed as such c^* for which $S(c^*)$ is maximal. This modification makes the method less local since all candidates of 3×3-neighborhood of the current node take part in the construction of estimate.

Note that $0 \le S(c) \le 8$ for the inner nodes of a regular grid; if the flow is continuous, the maximum (i.e., 8) must be reached with a high probability.

Hence, if max $S(c)<8$, the flow discontinuity is quite probable. In this case, we choose the estimate at random between the best candidates. The situation stays unclear for the rare case of several candidates with $S(c)=8$. Such observations lead to various simple schemes of choice of a single extremum. A study on the choice strategies based on $S(c)$ lies beyond the scope of this paper.

2.3 Dynamic Error

Note that our detector constructs the velocity estimate at the centre of domain B as the mean velocity inside B. In this section we apply elementary properties of Taylor series to find the deviation of the real velocity from the estimate.

Proposition. Let F be a vector function, $F(x) \in R^m$, of vector argument $x \in R^n$ that has all partial derivatives of the first and the second order. Let $\widetilde{F}(x_c) = \{\widetilde{F}_1, \widetilde{F}_2, ..., \widetilde{F}_m\}$ be a value constructed by averaging values of $F(x)$ in the n-dimensional cube of size $2r$ centered at x_c. Then for any component i=1, ..., m, the main term of decomposition of the error $\widetilde{F}_i - F_i(x)$ into Taylor series can be expressed as follows:

$$\widetilde{F}_i - F_i(\text{x}) \approx \left(\frac{\partial^2 F_i}{(\partial x_1)^2} + ... + \frac{\partial^2 F_i}{(\partial x_n)^2} \right) \frac{r^2}{6} . \tag{4}$$

The proof is trivial: the decomposition of $F_i(x)$ into Taylor series is

$$F_i(x) = F_i(x_c) + \sum_j \frac{\partial F_i}{\partial x_j}(x_j - x_{j,c}) +$$

$$\frac{1}{2} \sum_{j,k} \frac{\partial^2 F_i}{\partial x_j \partial x_k}(x_j - x_{j,c})(x_k - x_{k,c}) + ...,$$

where all derivatives are computed at x_c. Let us apply averaging (integration in the n-dimensional r-cube) to the right hand side:

$$\frac{1}{2^n r^n} \int \cdots \int_{x_1=x_{1,c}-r; ... x_n=x_{n,c}-r}^{x_1=x_{1,c}+r; ... x_n=x_{n,c}+r} [F_i(x_c) +$$

$$\sum_j \frac{\partial F_i}{\partial x_j}(x_j - x_{j,c}) +$$

$$\frac{1}{2} \sum_{j,k} \frac{\partial^2 F_i}{\partial x_j \partial x_k}(x_j - x_{j,c})(x_k - x_{k,c}) + ...] dx_1 .. dx_n,$$

and find the first non-zero term after the integration of this expression:

$$F_i(x_c) + \frac{1}{2^n r^n} \int \cdots \int_{x_1=x_{1,c}-r; ... x_n=x_{n,c}-r}^{x_1=x_{1,c}+r; ... x_n=x_{n,c}+r}$$

$$\frac{1}{2} \sum_j \frac{\partial^2 F_i}{(\partial x_j)^2}(x_j - x_{j,c})^2 dx_1 ... dx_n + ...$$

$$= F_i(x_c) +$$

$$\frac{1}{2} \frac{1}{2^n r^n} \sum_j \frac{\partial^2 F_i}{(\partial x_j)^2} (2r)^{n-1} \frac{1}{3}(x_j - x_{j,c})^3 \Big|_{x_j=x_{j,c}-r}^{x_j=x_{j,c}+r} + ...$$

$$= F_i(x_c) + \frac{1}{2} \frac{1}{2r} \sum_j \frac{\partial^2 F_i}{(\partial x_j)^2} \frac{2}{3} r^3 + ...$$

$$= F_i(x_c) + \frac{1}{6} r^2 \sum_j \frac{\partial^2 F_i}{(\partial x_j)^2} + ... \qquad \text{Q.E.D.}$$

The interesting detail is that the first partial derivatives and the mixed derivatives of the second order do not participate in Eq. (4).

This simple property can be used in two ways as follows.

The first way is that if the second partial derivatives at the centre of cube in the right-hand side of (2.2) are known a priori, then an enhanced estimate can be written as

$$\widetilde{F}_i^* = \widetilde{F}_i - \left(\frac{\partial^2 F_i}{(\partial x_1)^2} + ... + \frac{\partial^2 F_i}{(\partial x_n)^2} \right) \frac{r^2}{6} .$$

It compensates the main term of decomposition of the dynamic error in Taylor series.

The second way consists in a simple numeric method for computation of the deviation, that is, of $\| \widetilde{F}_i - F_i(\text{x}) \|$. For this goal, the derivatives in the right-hand side of (4) can be substituted with simple approximations:

$$\frac{\partial^2 F_i}{(\partial x_j)^2} \approx \frac{F_i(x_1, ..., x_j+r, ..., x_n) + F_i(x_1, ..., x_j-r, ..., x_n) - 2F_i(x)}{r^2},$$

after which the right-hind side of (2.2) can be rewritten as

$$\sum_j [F_i(x_1, ..., x_j + r, ..., x_n) +$$

$$+ F_i(x_1, ..., x_j - r, ..., x_n) - 2F_i(x)] / 6. \tag{5}$$

Given a dataset provided with ground truth, the expression (2.3) can be easily estimated.

3 Noise Generation

It is known that sensor noise in digital images is formed by components of different kinds (Horowitz and Hill [14]). Noise tolerance is studied here only with respect to Poisson noise, also known as *photon* or *shot* noise. Our attention is focused on weak signal images important, for example, for night vision systems or computer radiography, and for such images the Poisson noise overrides all other kinds.

Our scheme for generation of noise is described in the remainder of this section.

Given $b \in [0, 255]$ as an exact pixel value represented in the one-byte dynamic range, a variate to represent this pixel with noise is generated as $T(M_{\rho,W}^{-1}(P(M_{\rho,W}(b))))$. These variates are statistically independent for different pixels. The explanation follows.

$M_{\rho,W}$ and $M_{\rho,W}^{-1}$ are, respectively, a linear map $M_{\rho,W}$: $[0, 255] \to R_+$ and its inverse; $\mu = M_{\rho,W}(b)$ represents the expectation of the number of photons at the pixel photodetector. *P* represents a generator of variates [15]: each call of *P* with parameter $\mu \geq 0$ returns a variate $P(\mu)$, distributed by Poisson's law with expectation μ. *T* clamps big values: if $x \leq 255$ then $T(x)=x$, else $T(x)=255$. Parameter ρ stands for the level of noise and is defined as *standard_deviation/signal*, where the signal corresponds to a certain value *W* ('brightness of white') from the dynamic range: *W* must be as high as possible, but such that the need to truncate the value of $M_{\rho,W}^{-1}(P(M_{\rho,W}(b)))$ would have a specified small probability, otherwise it is difficult to interpret the experiments with noise. For instance, if *W*=210, then for any ρ<0.3 and *b*<210, the probability of truncation is <0.015%. In turn, the image area with $b \geq 210$ is small usually: for example, it is <2% for the Yosemite. Since for the Poisson distribution the mean value is equal to the variance, these definitions lead easily to equation

$$M_{\rho,W}(W)=1/\rho^2, \qquad (6)$$

which together with the condition $M_{\rho,W}(0)=0$ completely determines the maps $M_{\rho,W}$ and $M_{\rho,W}^{-1}$.

4 Experimental Results

Note that if a flow discontinuity line divides the square *B* of Section 2.1 into two halves, it is quite probable that the set of candidates generated by the 3FM includes velocities of either half. In this case, selection of a correct candidate cannot be done locally. A similar situation arises while applying our detector to datasets with layered flow or other violations of BCE. Processing such hard datasets requires some non-local techniques beyond the scope of this paper: our primary goal is to understand if our detector has merits for further study and, if yes, to combine it in future with the non-local techniques.

The majority of datasets in the Middlebury open evaluation system [3] are hard, except for the Yosemite one. According to the system specifications, we treated all evaluation datasets and submitted them to evaluation. Nevertheless, only the experiment with Yosemite is relevant to the stated goal.

The second goal is to study noise tolerance of the method. For this we make use of training datasets provided with ground truth, namely, some Middlebury training datasets and the Yosemite dataset by Michael Black [5]. Note that the Yosemite version of the evaluation system is slightly different and without publicly available ground truth.

We adopt main measures and statistics proposed by different authors [4, 23, 28], and summarized in [2]. Here we employ Endpoint Error (EE) and the robust accuracy measures EE50, EE75, and EE95 applied to EE. The EE is defined as

$$EE = \sqrt{(u - u_{GT})^2 + (v - v_{GT})^2},$$

where (*u, v*) and (u_{GT}, v_{GT}) are the flow estimate and the ground truth, respectively.

Then EE50, EE75, and EE95 represent robust accuracy measures applied to EE. The robust accuracy measures are defined as follows: let AX denote the accuracy of the error measure at the Xth percentile, after sorting the errors from low to high. So, for the flow errors EE, we compute A50, A75, and A95. They are denoted in our tables, respectively, as EE50, EE75, and EE95.

The number of scales (Section 2) is set to three in the experiments. The lowest scale varies from 4 to 5 pixels for different datasets, and the highest scale, from 17 to 21. The step between two adjacent profiles for computation of functionals for the 3FM (Section 2.1) is set to two pixels. The flow computation is performed first by our local detector on the nodes of a regular grid and then interpolated to the rest of pixels. The step between the nodes of the grid varies from 5 to 7 pixels depending on the size of frames of a dataset.

Table 1 shows distribution of the methods with respect to EE, as submitted to the evaluation system [3]. It takes into account the results obtained on the Yosemite datasets. Our method with EE=0.12 belongs to the best 20 methods of 87 submitted to the Middlebury system for evaluation. We believe that EE = 0.15÷0.2 for Yosemite corresponds to a reasonably high accuracy quite sufficient for most applications.

Table 2 summarizes our experiments for the noise tolerance study. Our results of the first line coincide with the corresponding data evaluated in [3].

Here we see how robust our method is under increasing noise. A change from zero noise to 1%-noise has little statistical effect. A further increase from 1% up to 10% deteriorates the parameters insignificantly for practical purposes. Note, however, that according to Eq. (3.1), a 10-fold increase of noise leads to a 100-fold diminution of signal (Table 3).

At ρ=0.2, noise influences the parameters notably stronger, although they remain at a reasonable level for Yosemite and Rubber_Whale, but not for Hydrangea (which is much harder). For this last dataset, one more experiment is presented for ρ=0.15 to localize better the limit of applicability of the method. For experiments with noise, the computed flow is shown in Fig.4. For representation of the computed flow there, we adopt the color-coding format. The coded images are constructed using the Middlebury open software [27].

We applied expression (2.3) to compute the mean dynamic error (DE) of the detector. Our results show that DE is negligible compared to EE, for example, for Yosemite DE=0.014 vs. EE=0.12.

The processing time for the described algorithm is about 1-2 sec/node on a conventional computer.

**Table 1.** Performance of our method (EE=0.12) vs. others for the Yosemite dataset

End-point Error (EE)	Number of methods with a better performance
0.07	1
0.10	4
0.11	11
0.12	20
0.13	29
0.14	37
0.20	68
0.59	89

Table 2. Dependence of main measures on the level of noise

Dataset	Noise (ρ)	EE	EE95	EE75	EE50
Yosemite (by Black, [5])	0	0.12	0.27	0.12	0.08
	0.01	0.12	0.27	0.12	0.08
	0.1	0.16	0.35	0.15	0.08
	0.2	0.3	0.6	0.2	0.09
Rubber Whale	0	0.72	0.54	0.18	0.08
	0.01	0.72	0.54	0.18	0.08
	0.1	0.82	0.78	0.21	0.11
	0.2	1.14	1.34	0.27	0.14
Hydrangea	0	1.4	1.55	0.55	0.06
	0.01	1.4	1.55	0.55	0.06
	0.1	1.72	1.62	0.6	0.09
	0.15	4.24	1.75	0.77	0.15
	0.2	6.7	2.6	1.08	0.4

Table 3. How signal depends on the noise level

Noise (ρ)	Signal: the number of photons at a bright_pixel
$3*10^{-5}$ (no noise)	$1.17*10^{10}$
0.01	$1.05*10^{5}$
0.1	1050
0.2	267
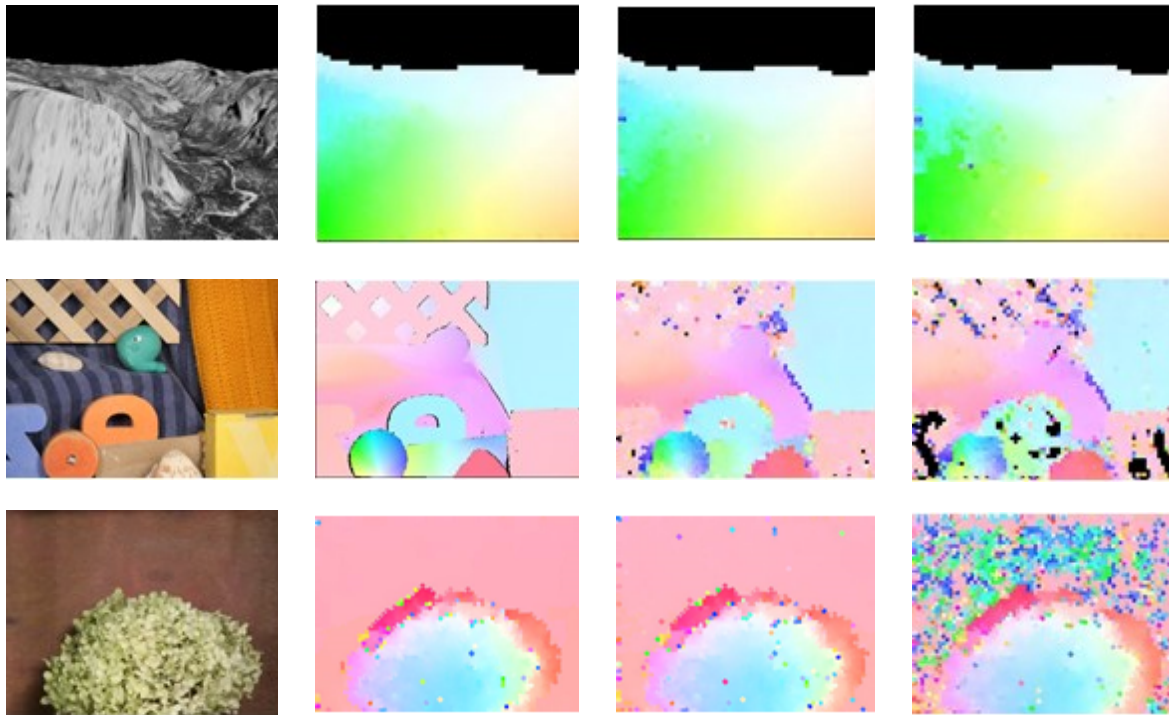
Fig. 4. Influence of noise on the computed flow. Top, middle, and bottom row represents, respectively, Yosemite, RubberWhale, and Hydrangea; the images in the left column are frames of the corresponding dataset; then from the left to right: the flow computed without noise, for ρ=0.1, and ρ=0.2

This assumes independent treating of each node. For this case the full treatment of Yosemite takes about 2000 sec. It is slow because of the brute force search for any scale and node. Nevertheless, the computation becomes 20-40 times faster, that is, about 70 sec for Yosemite, for a modified algorithm that takes advantage of initial guess for velocity estimates. This modification is simple for the areas with continuous flow, but it makes slightly more complex the detector logic at the flow discontinuities. Anyway, we believe that the processing time is not a critical factor because the kernel computations of our method fit ideally the parallel computing, which leads to the opportunity of a very fast and reasonably cheep implementation by the GPGPU [24].

5 Conclusion

A novel periodicity-based approach to computation of optical flow is presented.

The core algorithm is implemented as a local detector designed for independent applications at random locations of the image frame. The detector employs specific properties of the test for periodicity of Khachaturov [18, 19].

Here we study the detector in its genuine form, that is, without using the prior term. Even so, the evaluation by the Middlebury system [3] demonstrates a reasonably high accuracy of the detector for continuous flow; on the other hand, the detector is not competitive on hard datasets, which agrees with the arguments at the beginning of Section 4.

We believe that in future the performance can be improved furthermore by the prior term and combining our detector with some of the approaches mentioned in Section 1. However, the results presented in Section 4 allow us to claim that the chase for improving accuracy is not as important for applications as noise tolerance of the method. For example, the approach can be applied

to medical applications and more specifically to preventive tomography.

To clarify this assertion, note that a high quality of images produced by modern tomography equipment is aimed to allow a diagnostician to interpret a *single* tomographic image by a direct visual inspection. In contrast to that, preventive tomography suggests processing *several* tomographic images of a patient. All images should be taken under the same conditions with a certain period, say, one per year. A proper analysis of the images is completely automatic and it does not require any medical interpretation of individual images because the goal is to estimate their dynamics and, more specifically, to detect an illness at its early stage by suspicious changes in human body.

In this respect, the results presented in Section 4 show that automatic study of dynamics can be provided practically without any loss of performance by images of a much lower quality than that required for a static visual inspection. A direct consequence of our results for preventive tomography consists in the fact that, while taking images of a patient, the exposure can be several hundred times less than for conventional tomography. That is, harmful radiation exposure for preventive tomography becomes practically negligible and, in addition, generation of images is several hundred times faster than for conventional tomography – some seconds instead of dozens of minutes.

The presented results clearly determine the main priorities for future work: we believe that it must be focused on real applications that use weak signal images as the input, in particular, on night vision systems and preventive tomography. For the latter case, our processing scheme should be extended from 2D- to 3D-images and verified on temporal sequences of real tomographic images; an important part of research consists in medical interpretation of the computed flow. It must be provided in collaboration with experts in medicine.

Acknowledgement

References

1. **Anandan, P. (1989).** A computational framework and an algorithm for the measurement of visual motion. *IJCV*, Vol. 2, No.3, pp. 283–310.

2. **Baker, S., Scharstein, D., Lewis, J.P., Roth, S., Black, M. J., & Szeliski, R.** (**2011**). A Database and Evaluation Methodology for Optical Flow. *IJCV,* Vol. 92, No. 1, pp.1–31.

3. **Baker, S., Scharstein, D., Lewis, J.P., Roth, S., Black, M.J., & Szeliski, R. (2013).** *Middlebury open evaluation system* [online].

4. **Barron, J.L., Fleet, D.J., & Beauchemin, S. (1994).** *Performance of optical flow techniques. IJCV,* Vol. 12, No. 1, pp. 43–77.

5. **Black, M.J. (2013).** *The Yosemite dataset provided with ground truth* [online].

6. **Brox, T. & Malik, J. (2010).** Large displacement Optical Flow: descriptor matching in variational motion estimation, *IEEE TPAMI*, doi: 10.1109/TPAMI.2010.143.

7. **Bruhn, A., Weickert, J., & Schnorr, C. (2005).** Lucas/Kanade meets Horn/Schunck: Combining local and global optic flow methods. *IJCV*, Vol. 61, No. 3, pp. 211–231.

8. **Fischler, M. & Bolls, R. (1981).** Random sample consensus: A paradigm for fitting with application to image analysis and automated cartography. *Communication of the ACM*, Vol. 24, No. 6, pp. 381–395.

9. **Fleet, D.J. & Jepson, A.D. (1990).** Computation of component image velocity from local phase information. *IJCV*. Vol. 5, pp. 77–104.

10. **Goldluecke, B. & Cremers, D. (2010).** Convex Relaxation for Multilabel Problems with Product Label Spaces. *Proc. of European Conf. on Computer Vision (ECCV-2010)*.

11. **Heeger, D.J. (1987).** Model for the extraction of image flow. *J. Opt. Soc. Am.* A 4*,* pp. 1455–1471.

12. **Horn, B. (1986).** *Robot vision.* Cambridge: MIT Press.

13. **Horn, B. & Schunck, B.G. (1981).** Determining optical flow. *Artificial Intelligence,* Vol. 17, pp. 185–203.

14. **Horowitz, P., & Hill, W. (1989).** *The Art of Electronics.* 2nd edition. Cambridge (UK): Cambridge University Press.

15. **Hörmann, W., Leydold, J., & Derflinger, G. (2004).** *Automatic nonuniform random variate generation.* Springer.

16. **Jepson, A., & Black, M.J. (1993).** Mixture models for optical flow computation. *CVPR*, pp. 760–761.

17. **Jojic, N., & Frey, B. (2001).** Learning flexible sprites in video layers*. CVPR,* Vol. 1, pp. 199–206.

18. **Khachaturov, G. (1995).** An approach to detection of line elements. *Proc. of the Second Asian Conference on Computer Vision (ACCV'95)*, Vol. 3, pp. 559–563.

19. **Khachaturov, G. (2011).** A scalable, high-precision, and low-noise detector of shift-invariant image locations. *Pat. Rec. Letters*, Vol. 32, pp. 145–152, doi: 10.1016/j.patrec.2010.10.002.

20. **Lucas, B.D. & Kanade, T. (1981).** An iterative image registration technique with an application to stereo vision. *Proc. of DAPRA Imaging Understanding Workshop*, pp. 121–130.

21. **Marr, D. & Ullman, S. (1981).** Directional selectivity and its use in early visual processing. *Proc. Roy. Soc.,* London B 211, pp. 151–180.

22. **Nagel, H.H. (1989).** On a constraint equation for the estimation of displacement rates in image sequences*. IEEE Trans PAMI,* Vol. 11, pp. 13–30.

23. **Otte, M. & Nagel, H.H. (1994).** Optical flow estimation: advances and comparisons. *Proc. of the European conference on computer vision,* pp. 51–60.

24. **Owens, J., Luebke, D., Govindaraju, N., Harris, M., Krüger, J., Lefohn, A., & Purcell, T. (2007).** A Survey of General-Purpose Computation on Graphics Hardware. *Computer Graphics Forum,* Vol. 26, No. 1, pp. 80–113.

25. **Rao, C.R., Toutenburg, H., Fieger, A., Heumann, C., Nittner, T., & Scheid, S. (1999).** Linear Models: Least Squares and Alternatives. *Springer Series in Statistics.*

26. **Reddy, B.S. & Chatterji, B.N. (1996).** An FFT-based technique for translation, rotation, and scale-invariant image registration. *IEEE Trans. on Image Processing,* Vol. 5, No. 8*,* pp. 1266–1271.

27. **Scharstein, D. (2013).** *Open source code for the flow representation in the colour-coding format* [online].

28. **Scharstein, D. & Szeliski, R. (2003).** High-accuracy stereo depth maps using structured light. *Proc. of the IEEE conference on computer vision and pattern recognition*, pp. 195–202.

29. **Seitz, S., Curless, B., Diebel, J., Scharstein, D., & Szeliski, R. (2006).** A comparison and evaluation of multi-view stereo reconstruction algorithms. *Proc. of the IEEE conference on computer vision and pattern recognition.* Vol. 1, pp. 519–526.

30. **Singh. (1990).** An estimation theoretic framework for image-flow computation. *Proc. of ICCV*, Osaka, pp. 168–177.

31. **Sudderth, E., Sun, D., & Black, M. (2012).** Layered Segmentation and Optical Flow Estimation over Time. *CVPR,* doi: 10.1109/CVPR.2012.6247873.

32. **Tagliasacchi, M. (2007).** A Genetic Algorithm for Optical Flow Estimation. *Image and Vision Computing,* Vol. 25, pp. 141–147, doi: 10.1016/j.imavis.2006.01.021.

33. **Uras, S., Girosi, F., Verri, A., & Torre, V. (1988).** A computational approach to motion perception. *Biol. Cybern*., Vol. 60, pp. 79–87.

34. **Werlberger, M., Pock, T., & Bischof, H. (2010).** Motion estimation with non-local total variation regularization. *Proc. of IEEE Conference CVPR-2010.*

35. **Xu, L., Chen, J., & Jia, J. (2008).** A segmentation based variational model for accurate optical flow estimation. *Proc. of the ECCV-2008,* Vol. 1, pp. 671–684.

36. **Zimmer, H., Bruhn, A., Weickert, J., Valgaerts, L., Salgado, A., Rosenhahn, B., & Seidel, H.P. (2009).** Complementary optic flow. *Proc. of Int. Conf. on Energy Minimization Methods in Comp.Vis. and Pat.Rec. (EMMCVPR).*

Georgii Khachaturov recieved his Ph.D. in Technical Cybernetics from Leningrad Polytechnic Institute in 1982 and his Master Degree in Mathematics from Leningrad State University in 1973. During 1974-1992 he was with the Russian State Scientific Center for Robotics (Saint-Petersburg, Russian Federation). Since 1994 he is with UAM-Azcapotzalco, Mexico City, as professor and researcher.

Silvia Beatriz González Brambila recieved her Ph.D. in Computer Science from Instituto Tecnológico y de Estudios Superiores de Monterrey, Morelos, Mexico. Her Master and Bachelor degrees are from Universidad Autónoma Metropolitana. She works as a professor and researcher at the Systems Department of UAM, campus Azcapotzalco. She was coordinator of the Computer Science Master Degree Program,

coordinator of the Computer Engineering Program, a teaching coordinator at the Basic Sciences and Engineering Division in UAM, campus Azcapotzalco. She was also coordinator of the General Examination Degree Exit (EGEL) in CENEVAL.

Jesús Isidro González Trejo received his Ph.D. (Doctor in Sciences) from Universidad Autónoma Metropolitana, campus Iztapalapa, in 1998. He works as a professor and researcher at the Systems Department of UAM, campus Azcapotzalco, since 1996. Since 1998 he has been a member of National System of Researchers of Mexico where at present he is at Level II. His research interests are related to analysis, simulation and control of systems using dynamics of fluids, analysis of time series, and image processing.