Fast Optic Flow Computation with Discrete Wavelets

Christophe Bernard
Centre de Mathématiques Appliquées
Ecole Polytechnique, 91128 Palaiseau Cedex, France
email: bernard@cmapx.polytechnique.fr

May 12, 1997

Abstract

This paper describes a new way to compute the optical flow based on dyadic filtering and subsampling pyramids. It is based on the projection of the optical flow equation on vectors of a wavelet basis. This algorithm is thus of complexity $O(N)$ if one image of the sequence has N pixels, and opens the way to efficient and unexpensive optical flow computation. Features of this algorithm include multiscale treatment of time aliasing and estimation of illumination changes.

Keywords: Optical flow.

Résumé

Ce rapport technique décrit une nouvelle de calcul du flot optique d'une séquence d'images basée sur des pyramides de filtrage et de codage en sous-bande. Elle consiste à projeter l'équation du flot optique sur des vecteurs d'une base d'ondelettes. La complexité de cet algorithme est donc désormais en $O(N)$, où N est le nombre de points de chaque image de la séquence. Cette méthode ouvre la voie d'un calcul rapide et peu coûteux du flot optique. L'algorithme est capable par son approche multi-échelles de gérer l'aliasage temporel, et peut également mesurer des variations d'illumination.

Mots-clés: Flot optique.

1 Introduction

Optic flow detection is an already widely studied problem whose goal is to explain changes in an image sequence as a result of a motion field. Applications range from moving image compression to real scene analysis and robotics.

Given an image sequence $I_t(x,y)$, the optical flow (v_x, v_y) has to match the well known *optical flow equation*

$$\frac{\partial I_t}{\partial x} v_x + \frac{\partial I_t}{\partial y} v_y + \frac{\partial I_t}{\partial t} = 0 \tag{1}$$

No pointwise resolution of the optical flow equation is possible, since on each location and each time, this would consist in solving a single scalar equation for two scalar unknowns. This is the *aperture problem.*

1.1 Previous work

Horn & Schunck [13] [14] wrote a pioneering paper on the subject, in which they add a smoothness constraint. Region-matching methods [2] differential methods [14] [15] and spatiotemporal filtering methods [1] [6] [9] [11] [12] appeared, on which Barron *& al.* made an extensive review [3]. Later, Burns *& al.* developed a discrete wavelet spatiotemporal filtering technique [4], and Weber & Malik a filtered differential method [20].

The only method that relies on dyadic filtering and subsampling schemes is Burns & al.'s. However, their method relies on 3D wavelet transforms and requires large sequences of consecutive pictures. The approach described hereafter is a projected differential method, that relies on the analysis of 2 consecutive pictures only, and on 2D wavelet transforms. Consequences are: a much faster algorithm, that can even be used as such for disparity estimation in stereo vision.

1.2 Suggested solution

To get rid of this *aperture problem*, we suppose that we dispose of a discrete wavelet basis $(\psi^s_{jkk'})_{s\in S, j,k,k'\in\mathbb{Z}}$ of $L_2(\mathbb{R}^2)$, where

$$\psi^s_{jkk'} = 2^j \psi^s(2^j x - k, 2^j y - k')$$

j is a resolution index, (k,k') is a 2-dimensional translation index and S is a set of orientation indexes.

Example 1 *An example basis of $L_2(\mathbb{R}^2)$ that is widely used in image processing is, given a scaling function ϕ and a wavelet ψ in $L_2(\mathbb{R})$, the one based on the three following wavelet orientations:*

$$\psi^x(x,y) = \psi(x)\phi(y) \tag{2}$$
$$\psi^y(x,y) = \phi(x)\psi(y) \tag{3}$$
$$\psi^{xy}(x,y) = \psi(x)\psi(y) \tag{4}$$

Given such a basis, for each resolution j and location (k,k'), assuming that v_x and v_y are constant over the support of any $\psi^s_{jkk'}$, $s \in S$, we do an inner product

of (1) with these wavelets, obtaining after an integration by parts a set of $\#S$ equations

$$\left\langle I, \frac{\partial \psi^s_{jkk'}}{\partial x} \right\rangle v_x + \left\langle I, \frac{\partial \psi^s_{jkk'}}{\partial y} \right\rangle v_y = \frac{\partial}{\partial t} \left\langle I, \psi^s_{jkk'} \right\rangle \tag{5}$$

in two unknowns: v_x and v_y. We write them in a condensed way as the system

$$M_{jkk'} \begin{bmatrix} v_x \\ v_y \end{bmatrix} = Y_{jkk'} \tag{6}$$

In the example of the basis above, we obtain an overdetermined system of 3 equations. This approach is very close to Weber & Malik's [20]. However, we use in this paper a discrete set of measure functions in order to take advantage of the computational efficiency of dyadic filter banks, no time filtering is required, and thus no assumption on the time dependence of the optic flow is necessary.

1.3 Underlying assumption

The above equations hold only if the optical flow is nearly constant over the support of the wavelets. This assumption is not new. Indeed, all optical flow methods rely on an assumption on the space dependence of the optical flow.

Horn & Schunck suggest to find the most regular optical flow field matching this optical flow condition. The smoothness measure is the integral

$$V = \int \int (\nabla v_x)^2 + (\nabla v_y)^2 dxdy$$

Region based matching methods, as well as spatiotemporal filtering based methods [4] [6] [9] [11] [12] [20] that are local filtering methods always rely on the assumption that the optical flow is locally constant in space. The latter methods even rely on the assumption that the optical flow is locally constant in time.

1.4 Road map

In this paper, we will respectively focus on time aliasing (section 2) to find out at which scale a given displacement can be measured. In section 3, we will describe the coarse to fine propagation. In section 4, we will show how to design scale separable wavelets, so that the corresponding inner products that are the coefficients of the local systems can be computed with pyramid filtering and subsampling schemes. Section 5 will be devoted to numerical experimentation.

2 Time aliasing

The image sequence being sampled in time, the time derivatives of inner products like $\int \int I_t(x,y)\psi(x,y)dxdy$ have to be estimated with finite differences.

The error of such estimations is high if the displacement (v_x, v_y) between two pictures I_t and I_{t+1} is large. This phenomenon has been pointed out by many authors, and a multiscale approach is generally considered as the way to solve this problem [2] [4] [17] [20].

2.1 Time aliasing error estimation

Examining the one-dimensional case, with a pattern I moving with velocity v, we see indeed that the finite difference estimate

$$\frac{\partial}{\partial t} \int I_t(x)\psi(x)dx \simeq \int \left(I_{t+1}(x) - I_t(x)\right) \psi(x)dx$$

is valid for any image I iff

$$v\frac{\partial \psi}{\partial x} \simeq \psi(x+v) - \psi(x)$$

or in Fourier domain

$$iv\xi\hat{\psi}(\xi) \simeq (e^{iv\xi} - 1)\hat{\psi}(\xi)$$

Thus, the spectrum of ψ has to be included in an interval where approximation

$$iv\xi \simeq e^{iv\xi} - 1 \tag{7}$$

is valid. For a 15% error, the spectrum of ψ has to be in $[-0.3/v, 0.3/v]$

We see that bigger displacements can only be detected with ψ measure function whose spectrum is narrow, and that are thus highly space correlated. As a consequence, the space dependence of such measures must be coarser than for smaller displacements. This is a general limitation for all linear filtering and frequency based methods.

2.2 Higher order derivative approximation

A higher order finite difference estimate reduces significantly the impact of time aliasing. It consists in estimating the flow in $t + 1/2$ with

$$\begin{aligned} I_{t+1/2} &\simeq \frac{I_t + I_{t+1}}{2} \\ \frac{\partial}{\partial t} I_{t+1/2} &\simeq I_{t+1} - I_t \end{aligned}$$

This time, the spectrum of ψ has to be included in an interval where approximation

$$iv\xi \simeq 2\frac{e^{iv\xi/2} - e^{-iv\xi/2}}{e^{iv\xi/2} + e^{-iv\xi/2}} \tag{8}$$

is valid. For a 15% relative error, this interval is $[-1.23/v, 1.23/v]$, and is significantly larger than the previous one. Relative errors of approximations (7) and (8) are compared in figure 1

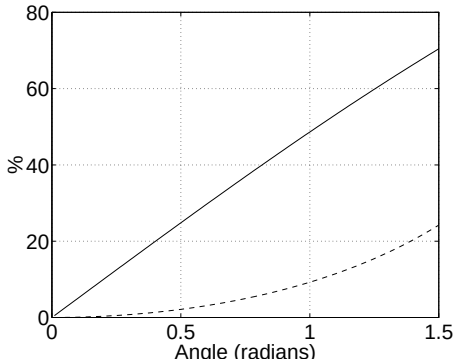


Figure 1: Compared relative errors (solid for (7), dashed for (8))

3 Flow estimation strategy

The flow estimation process is performed in a coarse to fine refinement. Coarser scale measurements are less subject to time aliasing, but their space dependence is also coarser. The algorithm should therefore try to rely as much as possible on finer scale measurements as long as

- the space frequency content of the local texture allows it
- the real displacement is lower than the limit imposed by aliasing.

3.1 Solving an overconstrained system

The systems we build with our multiscale measures

$$M_{jkk'} X = Y_{jkk'}$$

are always overdetermined. This has two advantages: ensuring higher stability to the flow estimation and giving a hint as whether the underlying assumption of uniform translation is correct.

An overconstrained system at a given resolution $M_{jkk'} X = Y_{jkk'}$ can be solved in the least square sense. If $M_{jkk'}$ and $Y_{jkk'}$ are complex matrices, the corresponding LS system is

$$M^{LS}_{jkk'} X = Y^{LS}_{jkk'} \tag{9}$$

where

$$M^{LS}_{jkk'} = (\Re M_{jkk'})^T \Re M_{jkk'} + (\Im M_{jkk'})^T \Im M_{jkk'} \tag{10}$$

$$Y^{LS}_{jkk'} = (\Re M_{jkk'})^T \Re Y_{jkk'} + (\Im M_{jkk'})^T \Im Y_{jkk'} \tag{11}$$

3.2 Core algorithm

This suggests therefore to use the following strategy, based on measurements $M_{jkk'}$ and $Y_{jkk'}$, $k \in \{1, \ldots, N_x 2^{-j}\}$ and $k' \in \{1, \ldots, N_y 2^{-j}\}$, that is on computation of coefficients $< \partial I_t/\partial t, \psi^s_{jkk'} >$, $< I_t, \partial \psi^s_{jkk'}/\partial x >$ and $< I_t, \partial \psi^s_{jkk}/\partial y >$.

1. *for each* $j = -j_{min}, .., 0$, *for each* $k = 1..2^{-j}x_{\max}$, $k' = 1..2^{-j}y_{\max}$, *if estimation at this location has been forbidden because of aliasing detected at a coarser scale, go to 2. Else,*

   (a) *if the LS system is not well conditioned, ie*

   $$\det M^{LS} < \theta_1 (\mathrm{tr} M^{LS})^3$$

   add LS constraints from the next coarser scale:

   $$\begin{aligned} M^{LS} &\leftarrow M^{LS} + \theta_3 M^{LS}_{j-1,k/2,k'/2} \\ Y^{LS} &\leftarrow Y^{LS} + \theta_3 Y^{LS}_{j-1,k/2,k'/2} \end{aligned}$$

   (b) *if the LS system is still not well conditioned, transfer coarser scale estimations, if any:*

   $$X_{jkk'} \leftarrow X_{j-1,k/2,k'/2} \tag{12}$$

   and go to **alias-check**.

   (c) *compute* $X = (M^{LS}_{jkk'})^{-1} X^{LS}_{jkk'}$.

   (d) *if the bias* $\|M_{jkk'} X - Y_{jkk'}\|$ *is higher than* $\theta_2 \|Y_{jkk'}\|$, *reject the measure: transfer coarser scale estimations as in (12) and go to* **alias-check**.

   (e) *if the measured displacement is higher than* α *times the grid step, reject the measure: transfer finer scale estimation, and forbid finer scale estimation at this location and go to* **end** (aliasing prohibits flow estimation at this scale and thus on finer scales).

   (f) *else accept the measure: set* $X_{jkk'} = X$.

   alias-check:

   (g) *if the measurement displacement* (v_x, v_y) *is higher than* $\alpha/2$ *times the grid step, we keep the measure, but forbid finer scale estimation at this location. Go to* **end**.

2. *transfer coarser scale estimation as in (12), and also forbid finer scale estimation (at this location).*

end:

3. *(end of the loop in* j, k *and* k'*).*

Remark 1 Forbidding finer scale estimation *when at scale* j *and location* (k, k') *consists in prohibiting estimations for* $J = j + 1$ *and* $K = 2k - 1 \ldots 2k$ *and* $K' = 2k - 1 \ldots 2k$. *This is propagated recursively through the scales by the algorithm above.*

4 Wavelet design

4.1 Scale separability

We want here *scale separable* wavelets, ie wavelets ψ^s matching equations like

$$\hat{\psi}^s(\xi,\eta) = \prod_{j=1}^{+\infty} m_j\left(\frac{\xi}{2^j},\frac{\eta}{2^j}\right) \tag{13}$$

because then, computing coefficients like $< I, \psi^s_{jkk'} >$ can be done with a fast filtering and subband coding scheme [7] [16].

We will show in this section how to have scale separable analytic wavelets (and why we need such wavelets), and that derivatives of scale separable functions are also scale separable. All our 2D wavelets will be tensor products of 1D functions thus also separable in x and y.

4.2 Extinction and analytic wavelets

If we use classical 2D real valued wavelets defined as tensor products (equations 2–4) of one dimensional real wavelets ϕ and ψ whose spectra are displayed in (2.c-d), numerical experimentation gives very poor and unstable results. A one dimensional approach shows that such a velocity estimation consists in writing

$$v(x_0) \simeq \frac{\frac{\partial}{\partial t}\int I_t(x)\psi(x-x_0)dx}{\int I_t(x)\psi'(x-x_0)dx} \tag{14}$$

If ψ is real valued, the denominator and the numerator in (14) both oscillate like cosine functions in x_0. The relative error can thus be very high when the denominator vanishes. We call this *extinction* after a similar phenomenon in light interference in physics.

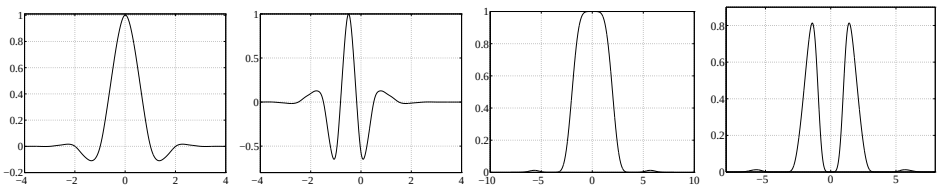
If however we replace ψ with its analytic part

$$\psi^+ = \frac{\psi + h * \psi}{2} \tag{15}$$

where $h(x) = 1/\pi x$ is the Hilbert filter, the denumerator and numerator in (14) will now oscillate like complex exponential funtions when x_0 varies, and thus be of almost constant modulus. The relative error will therefore keep low *at any time.*

Analytic measure functions are also used in spatiotemporal filtering techniques, where velocity tuned filters are analytic [9]. Note, however, that the Hilbert transform is also used to make filters direction selective and not analytic [4] [19]. Psychophysical evidence also supports the use of analytic wavelets. Daugman [8] identified a pair of (real valued) Gabor filters with a $\pi/2$ phase shift between them

$$\begin{aligned} f_1 &= e^{-(X-X_0)^2/2\sigma}\cos k.X \\ f_2 &= e^{-(X-X_0)^2/2\sigma}\sin k.X \end{aligned}$$



(a) ϕ (b) ψ (c) $|\hat{\phi}|$ (d) $|\hat{\psi}|$

Figure 2: ϕ, ψ and their Fourier transforms

Such a pair can equivalently be seen as a single complex filter

$$f = e^{-(X-X_0)^2/2\sigma} e^{ik.X} \tag{16}$$

that now has a non-symmetric spectrum, and is thus an approximation of an analytic transform of f_1. We will show in the next subsection how to design such a function in a filtering and subband coding framework, without using a costly Hilbert transform.

4.3 One dimensional analytic wavelet design

We can start from any filter pair m_0 and m_1 defining a scaling function and a wavelet as

$$\hat{\phi}(\xi) = \prod_{j=1}^{+\infty} m_0\left(\frac{\xi}{2^j}\right) \tag{17}$$

$$\hat{\psi}(\xi) = m_1\left(\frac{\xi}{2}\right)\hat{\phi}\left(\frac{\xi}{2}\right) \tag{18}$$

ϕ, ψ and their Fourier transforms are displayed in (2.a-d), for filters

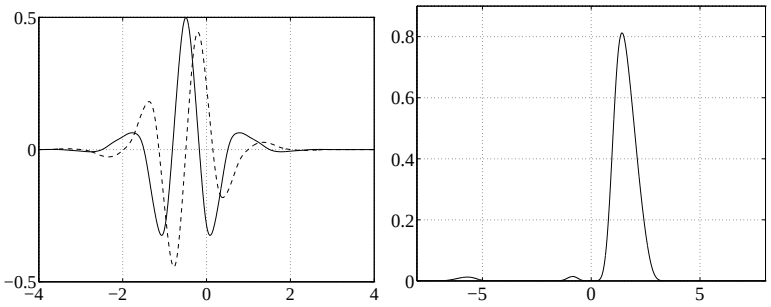
$$\begin{aligned} m_0(\xi) &= \frac{3\cos 5\xi - 25\cos 3\xi + 150\cos\xi + 128}{256} \\ m_1(\xi) &= m_0(\xi+\pi) \end{aligned}$$

If m_2 is a Deslauriers-Dubuc interpolation filter, such that

$$m_2(\xi) + m_2(\xi+\pi) = 1$$

then $\hat{\psi}^{\#}(\xi) = \hat{\psi}(\xi)m_2(\xi/2 - \pi/4)$ is a good approximation of $\hat{\psi}^{+}(\xi)$, since most of the negative frequency peak of ψ is cancelled by a vanishing $m_2(\xi)$. Again, $\psi^{\#}$ is displayed in figure 3 for $m_2 = m_0$. The remaining negative frequency content of $\psi^{\#}$ is not 0, but is less than 2% of $\psi^{\#}$'s total L_2 norm.

Thanks to the way $\psi^{\#}$ is defined, inner products $\int I(x)\psi^{\#}(x)dx$ can be computed like $\int I(x)\psi(x)dx$ with a filtering and subband coding scheme with a single additional discrete filtering step.



(a) $\psi^{\#}$, real and imaginary parts (b) $|\widehat{\psi^{\#}}|$

Figure 3: Approximation $\psi^{\#}$ of ψ^{+} and its Fourier transform

4.4 Derivative wavelets and corresponding filters

If a function f is an infinite convolution of discrete filters

$$\hat{f}(\xi) = \prod_{j=1}^{+\infty} M_j\left(\frac{\xi}{2^j}\right)$$

which is the case of ϕ and $\psi^{\#}$ defined above, a corresponding filter sequence to define f' is the following:

$$\begin{array}{rcll} N_j(\xi) & = & \frac{M_j(\xi)}{e^{i\xi}+1} & \text{if } j \geq 2 \\ N_1(\xi) & = & (e^{i\xi}-1)N_1(\xi) & \end{array}$$

because then

$$\prod_{j=1}^{+\infty} N_j\left(\frac{\xi}{2^j}\right) = i\xi\hat{f} = \widehat{f'}$$

Thus derivatives of scale separable functions are scale separable, which will be very useful for our purpose.

4.5 Two-dimensional analytic wavelets

Two dimensional wavelet shapes are defined as tensor products of one dimensional wavelets. Thus, we define ψ^s, $s = 1 \ldots 5$, as

$$\begin{aligned}
\psi^1(x,y) &= \psi^{\#}(x)\phi(y) \\
\psi^2(x,y) &= \phi(x)\psi^{\#}(y) \\
\psi^3(x,y) &= \psi^{\#}(x)\psi^{\#}(y) \\
\psi^4(x,y) &= \psi^{\#}(x)\overline{\psi^{\#}(y)} \\
\psi^5(x,y) &= \phi(x)\psi(y)
\end{aligned}$$

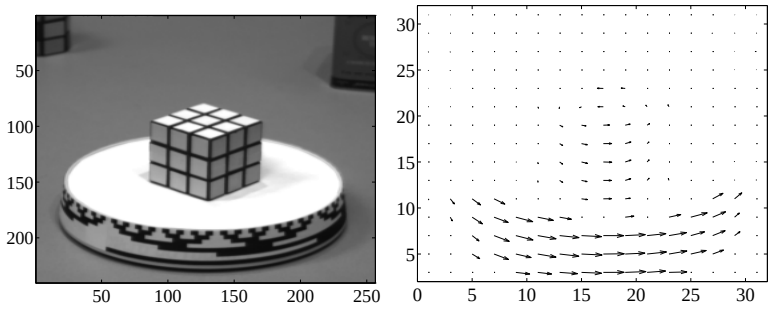
The same way the partial derivatives of ψ^s, $s = 1, \ldots 5$ can be written as tensor products of scale separable one dimensional functions defined in subsections 4.3 and 4.4.

5 Numerical experimentation

The algorithm was implemented as a matlab script, and run on several picture sequences from Barron *& al.*'s.

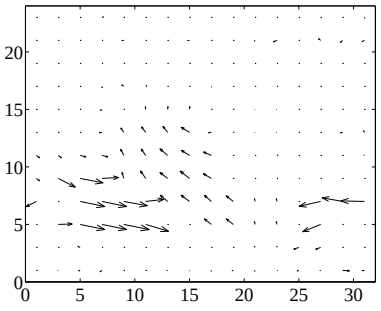
5.1 True sequences

Image sequences were downloaded from Barron *& al.*'s ftp site at *csd.uwo.ca*. The algorithm was tested on the rubik sequence (a rubik's cube on a rotating plate), the taxi sequence (where three vehicles are moving respectivement towards East, West and Northwest) and the nasa sequence, which a is zoom on a Coke can.

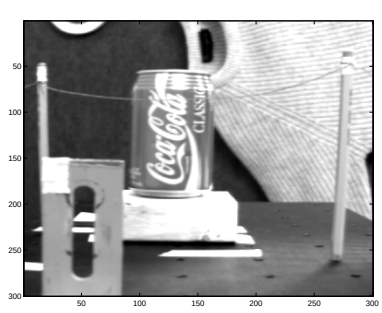


(a) rubik image (b) rubik flow

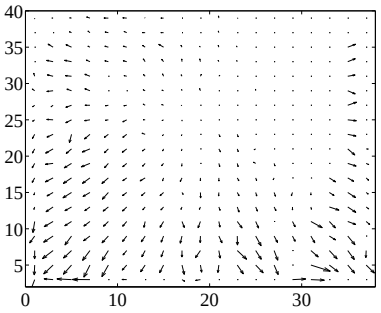
(c) taxi image



(d) taxi flow



(e) nasa image



(f) nasa flow

5.2 Synthetic sequences

Again sequences provided by Barron & al were used to check the new algorithm. Comparative results are provided on table 1. Numerical results are always given for the whole picture size, unless otherwise stated.

The main weakness of the approach suggested in this paper is "border effect" that reduces accuracy of the optic flow estimation near the boundaries. Yosemite estimation was therefore cropped of 5% (resp. 6.25%) of the total width (height) of the image, at each border to get the "cropped" error estimation displayed in table. Note that the density is then related to the total surface of the original pictures.

Sequences	Authors	Estimation error		Density
		average	std. dev.	
Transl. tree	Heeger	4.53	2.41	57.8 %
	Fleet & Jepson ($\tau = 2.5$)	0.32	0.38	74.5 %
	Fleet & Jepson ($\tau = 1.0$)	0.25	0.21	26.8 %
	Weber & Malik	0.49	0.35	96.8 %
	Bernard	1.42	0.53	52.6 %
Yosemite	Heeger	10.51	12.11	15.2 %
	Fleet & Jepson ($\tau = 2.5$)	4.25	11.34	34.1 %
	Fleet & Jepson ($\tau = 1.25$)	5.28	14.34	30.6 %
	Weber & Malik	4.31	8.66	64.2 %
	Bernard	9.6	11.9	47.3 %
	Bernard (borders cropped)	7.2	7.2	41.3 %

Table 1: Error and estimation density comparison of several methods

Method	Frames
Burns & al	64
Fleet & Jepson	21 or 15
Weber & Malik	10
Bernard	2

Table 2: Number of frames required to compute an optical flow map

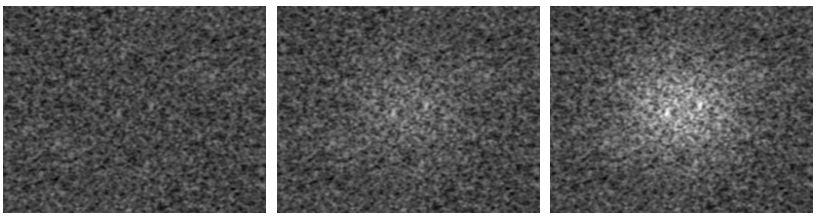
5.3 Illumination estimation

Using a new optical flow equation

$$\frac{\partial I_t}{\partial x} v_x + \frac{\partial I_t}{\partial y} v_y + \frac{\partial I_t}{\partial t} = \lambda I_t$$

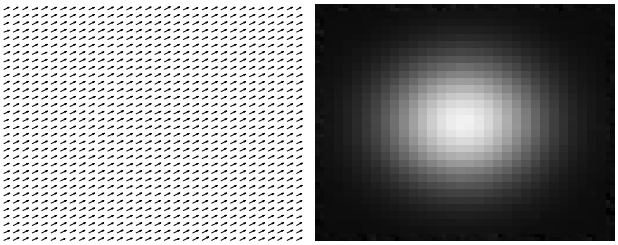
where λ is the logarithmic derivative of the illumination factor, using an additional wavelet shape $\psi^0(x,y) = \phi(x)\phi(y)$ of nonzero integral, we can now perform illumination change measurements, ie estimate a new unknow parameter λ.

A synthetic sequence (moving white noise, with increasing illumination of gaussian shape) was created. Three pictures of the sequence (figure 4) and the corresponding measured flow and illumination map (figure 5) are displayed.



(a) image 1 (b) image 2 (c) image 3

Figure 4: Moving random pattern with varying illumination



(a) flow (b) light log. derivative

Figure 5: Measured flow and illumination change

Conclusion

The method presented in this paper is an improvement of the existing ones in terms of reduced computational complexity. This reduction is gained because

- the algorithm is completely time-local. Only two frames are required to compute an optical flow map (see table 2).
- the pyramid filtering and subsampling scheme structure allows to measure displacements at several scales without massive convolutions. As an example, flow estimations (translating tree) were performed with a matlab script in less than 20 seconds on an HP-9000 workstation.

Aknowledgements

The author would like to express his gratefulness to Stéphane Mallat for very helpful comments and discussions and Jean-Jacques Slotine for stimulating discussions on possible applications in robotics.

References

[1] E. H. Adelson and J. R. Bergen, "Spatiotemporal Energy Models for the Perception of Vision," *J. Opt. Soc. Amer.*, Vol A2, pp. 284-299, 1985.

[2] P. Anandan, "A Computational Framework and an Algorithm for the Measurement of Visual Motion," *International Journal of Computer Vision*, Vol. 2, pp. 283-310, 1989.

[3] J.L. Barron, D.J. Fleet and S.S. Beauchemin, "Performance of Optical Flow Techniques," *International Journal of Computer Vision*, Vol. 12:1, pp. 43-77, 1994.

[4] T.J. Burns, S.K. Rogers, D.W. Ruck and M.E. Oxley, "Discrete, Spatiotemporal, Wavelet Multiresolution Analysis Method for Computing Optical Flow," *Optical Engineering*, Vol. 33:7, pp. 2236-2247, 1994.

[5] P.J. Burt and E.H. Adelson, "The Laplacian Pyramid as a Compact Image Code," *IEEE. Trans. Communications*, Vol. 31, pp. 532-540, 1983.

[6] C.W.G. Clifford, K. Langley and D.J. Fleet, "Centre-Frequency Adaptive IIR Temporal Filters for Phase-Based Image Velocity Estimation," *Image Processing and its Applications*, Vol. 4-6, pp. 173-177, 1995.

[7] I. Daubechies, *Ten Lectures on Wavelets*, Society for Industrial and Applied Mathematics, Philadelphia, 1992.

[8] J. G. Daugman, "Complete Discrete 2-D Gabor Transforms by Neural Networks for Image Analysis and Compression," *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. 36:7, pp. 1169-1179, 1988.

[9] D.J. Fleet and A.D. Jepson, "Computation of Component Image Velocity from Local Phase Information," *International Journal of Computer Vision*, Vol. 5, pp. 77-104, 1990.

[10] W.T. Freeman and E.H. Adelson, "The Design and Use of Steerable Filters," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 13:9, pp. 891-906, 1991.

[11] M. Gökstorp and P-E. Danielsson, "Velocity Tuned Generalized Sobel Operators for Multiresolution Computation of Optical Flow," *IEEE*, pp. 765-769, 1994.

[12] D.J. Heeger, "Optical Flow Using Spatiotemporal Filters," *International Journal for Computer Vision*, Vol. 1, pp. 279-302, 1988.

[13] B.K.P Horn and B.G. Schunck, "Determining Optical Flow," *A.I. Memo No. 572, Massachusetts Institute of Technology*, 1980.

[14] B.K.P Horn and B.G. Schunck, "Determining Optical Flow," *Artificial Intelligence*, Vol. 17, pp. 185-204, 1981.

[15] B. D. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision," *Proc. DARPA Image Understanding Workshop*, pp. 121-130, 1981.

[16] S.G. Mallat, "A Theory for Multiresolution Signal Decomposition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 11:7, pp. 674-693, 1989.

[17] E.P. Simoncelli, W.T. Freeman, "The Steerable Pyramid: a Flexible Architecture for Multi-Scale Derivative Computation," *2nd Annual IEEE International Conference on Image Processing, Washington DC*, 1995.

[18] A.B. Watson, "The Cortex Transform: Rapid Computation of Simulated Neural Images," *Computer Vision, Graphics, and Image Processing*, Vol. 39:3, pp. 311-327, 1987.

[19] A.B. Watson and A.J. Ahumada, Jr, "Model of Human Visual-Motion Sensing," *Journal of Optical Society of America*, Vol. A:2-2, pp. 322-342, 1985.

[20] J. Weber and J. Malik, "Robust Computation of Optical Flow in a Multi-Scale Differential Framework," *International Journal of Computer Vision, Vol. 14:1, pp. 5-19, 1995*