社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2007－CVIM－157 (16)
2007／1／12

Dagstuhl 算法によるオプティカルフロー計算

大西 直哉† 井宮 淳††

† 千葉大学 自然科学研究科
†† 千葉大学総合メディア基盤センター
〒263-8522 千葉市稲毛区弥生町 1-33

あらまし 本論文では, 多重解像度解析によるオプティカルフロー計算法が, 任意の低解像度画像対から計算をはじめて, 鮮明な画像対から計算されるオプティカルフローに常に収束することを証明する. また, 画像列の解像度が徐々に高くなるとき, 多重解像度解析を利用して, 鮮明な最終画像のオプティカルフローを計算できることを示す. このアルゴリズムを利用すると, 不鮮明な画像から初めて徐々に解像度を高くして目的の物体の動きによるオプティカルフローを計算できることになる. 提案するアルゴリズムは工学的な応用だけでなく, 人間が動きを追跡する仮定をモデル化したアルゴリズムにもなっている. 本論文は, Dagstuhl seminer の working group において, Reinhard Klette によって提示され, Leo Dorst, Atsushi Imiya が議論をした問題に対する解答である.

Dagstuhl Algorithm for Optical-Flow Computation

Naoya OHNISHI† and Atsushi IMIYA††

†School of Science and Technology, Chiba University
††Institute of Media and Information Technology, Chiba University
Yayoi-cho 1-33, Inage-ku, Chiba, 263-8522, Japan

Abstract In this paper, we introduce a class of multi-resolution optical flow computation algorithms. The Lucus-Kanade method with the pyramid transform (the LKP in abbreviation) is a promising method of the optical flow computation. This algorithm is a combination of variational method and multi-resolution analysis of images. There are possible extentions to the LKP. The first one is to adopt the other optical-flow computation method in each layer. For instance, we can adopt the Horn-Schunck method, the Nagel-Enkelmann method, correlation method, and block-matching method. The second extention is to compute optical flow from a pair of different resolution images. If the resolution of each image is higher than the previous one, the resolution of images in a sequence increases. We address the second extention of the Lucus-Kanade method with the pyramid transform.

1 Introduction

In this paper, we introduce a class of multi-resolution optical flow computation algorithms. The basic concept of the algorithms was formulated during Dagstuhl seminar by Reinhard Klette, Leo Dorst, and Atsushi Imiya in a working group meeting. An early version of the algorithm is mathematically formulated to answer the following question.

> If the resolution of images in a sequence is increasing, is it possible to compute optical flow?

asked by Reinhard Klette. The practical motivation of the question comes from the following problem.

> Starting from a low resolution image and increasing the resolution, the algorithm allows us to compute small and fast displacements of an object in a region of interest very fast.

Beyond engineering applications, the answer of the question might clarify a relation between motion cognition and focusing to a attention field. For instance, we human being see a moving object in a scene which is an observation of the environment around us. If we realise a moving object is important for cognition of the environment, we try to attend on the object and start to watch it by increasing the resolution to it locally.

The Lucus-Kanade method with the pyramid transform (the LKP in abbreviation) is a promising method of the optical flow computation. This algorithm is a combination of variational method and multi-resolution analysis of images. The main part of the LKP is to use optical-flow computation in a low reso-

lution layer to an initial estimation of the flow vectors of the higher resolution layer. For the LKP, the pyramid transform is applied to each of a pair of successive images in an image sequence to derive low resolution images. Since the image sizes of low resolution images are smaller than the sizes of the original images, it is easy to compute optical flow for these images. The result of computation is, however, an approximation solution. Therefore, this approximation solution is used to as initial data to refine optical flow for the lower resolution image pair.

There are possible extensions to the LKP. The first one is to adopt the other optical computation method in each layer. For instance, we can adopt, the Horn-Schunck method, the Nagel-Enkelmann method, correlation method, and block-matching method [1]. The second extension is to compute optical flow from a pair of different resolution images. If the resolution of each image is higher than the previous one, the resolution of images in a sequence increases. In this paper, we address the second extension.

Section 2 sumerises the basic properties on optical-flow computation. In section 3, we briefly describe some classes of T_σ. In section 4, we introduce the basic algorithms and a variation when T_σ is the classical regular pyramid transform. In section 4, we show some numerical results for our two new algorithms. Comparative studies with the traditional LKP are also given. In section 5, we discuss mathematical properties of our algorithms.

2 Optical Flow and Regularisation

Setting $f(\boldsymbol{x}-\boldsymbol{u}, t+1)$ and $f(\boldsymbol{x}, t)$ to be the images at time $t+1$ and t, the small displacement $\boldsymbol{u}$ of each point $\boldsymbol{x}$ is called optical flow of the image f.

For a spatio-temporal image $f(\boldsymbol{x}, t)$, $\boldsymbol{x} = (x, y)^\top$ the total derivative is given as

$$\frac{d}{dt}f = \frac{\partial f}{\partial x}\frac{dx}{dt} + \frac{\partial f}{\partial y}\frac{dy}{dt} + \frac{\partial f}{\partial t}\frac{dt}{dt} \tag{1}$$

where $\boldsymbol{u} = (\dot{x}, \dot{y})^\top = (\frac{dx}{dt}, \frac{dy}{dt})^\top$ is the motion $\boldsymbol{u} = \dot{\boldsymbol{x}} = (\dot{x}, \dot{y})^\top$ of each point $\boldsymbol{x} = (x, y)^\top$. Optical flow constrain [2, 3, 4]

$$\frac{d}{dt}f = 0 \tag{2}$$

implies that the motion of the point $\boldsymbol{u} = (\dot{x}, \dot{y})^\top$ is the solution of the singular equation,

$$f_x\dot{x} + f_y\dot{y} + f_t = 0. \tag{3}$$

Setting $f(\boldsymbol{x}-\boldsymbol{u}, t+1)$ and $f(\boldsymbol{x}, t)$ to be the images at time $t+1$ and t, for the small displacement $\boldsymbol{u}$ of each point $\boldsymbol{x}$, we have

$$f(\boldsymbol{x}-\boldsymbol{u}, t+1) - f(\boldsymbol{x}, t) \cong \nabla f^\top \boldsymbol{u} + f_t = \frac{d}{dt}f. \tag{4}$$

Therefore, we accept solution of eq. (2) as $\boldsymbol{u}$

To solve this equation, the regularisation method is employed

$$\int_{\mathbf{R}^2} |f_x\dot{x} + f_y\dot{y} + f_t|^2 dxdy + \alpha H(\dot{x}, \dot{y})dxdy, \tag{5}$$

where $H(x, y)$ is an appropriate positive bilinear function of x and y.

If $H(\cdot, \cdot)$ is in the form [3]

$$H(\dot{x}, \dot{y}) = tr(\nabla\dot{x}^\top \boldsymbol{A}\nabla\dot{x} + \nabla\dot{y}^\top \boldsymbol{A}\nabla\dot{y}), \tag{6}$$

where $\boldsymbol{A}$ is positive definite such that $0 < \rho(\boldsymbol{A}) \leq 1$, the Euler-Lagrange Equation of the energy function eq. (5) is

$$\nabla^\top \boldsymbol{A}\nabla\boldsymbol{u} = \frac{1}{\alpha}(\nabla_t f^\top \boldsymbol{v})\nabla f, \tag{7}$$

for $\boldsymbol{v} = (\boldsymbol{u}^\top, 1)^\top$ and $\nabla_t f = (\nabla f^\top, f_t)^\top$. Therefore, the embedding of the Euler-Lagrange equation to the PDE is

$$\frac{\partial}{\partial\tau}\boldsymbol{u} = \nabla^\top \boldsymbol{A}\nabla\boldsymbol{u} - \frac{1}{\alpha}(\nabla_t f^\top \boldsymbol{v})\nabla f. \tag{8}$$

If $\boldsymbol{A} = \boldsymbol{I}$, we have the Horn-Schunck regulaiser. Furthermore, the Nagel-Enklemann [3, 4] regularisation term for optical flow detection is expressed as

$$\boldsymbol{A} = \frac{1}{|\nabla f|^2 + 2\lambda^2}(\nabla f^\perp \nabla f^\top + \lambda^2 \boldsymbol{I}). \tag{9}$$

Setting $\mathbf{R}^2 = \bigcup_{i,j=1}^{m,n} \mathbf{D}(\boldsymbol{x}_{ij})$ to be a non-overlapping decomposition of $\mathbf{R}^2$, we call $\boldsymbol{x}_{ij}$ the seed of decomposition. If the flow vector is picewise-constant, that is, $\boldsymbol{u} = c$ in the interior each $\mathbf{D}(\boldsymbol{x}_{ij})$, the constraint define by eq. (6) is zero. Therefore, setting

$$L_{ij}(\boldsymbol{u}) = \int\int_{\mathbf{D}(\boldsymbol{x}_{ij})} |\nabla f^\top \boldsymbol{u} + f_t|^2 d\boldsymbol{x}. \tag{10}$$

we have the minimisation problem

$$L_{ij}(\boldsymbol{u}) \to \min \ 1 \leq i \leq m, \ j \leq j \leq n. \tag{11}$$

For each $L_{ij}(\boldsymbol{u})$, we have the relation

$$L(\boldsymbol{u}) = \sum_{i,j=1}^{m,n} \boldsymbol{v}_{ij}^\top \boldsymbol{S}_{ij}\boldsymbol{v}_{ij} \tag{12}$$

where

$$\boldsymbol{S}_{ij} = \int\int_{\mathbf{D}(\boldsymbol{x}_{ij})} \nabla_t f \nabla_t f^\top d\boldsymbol{x} \tag{13}$$

for $\nabla_t f = (f_x, f_y, f_t)^\top$.

Therefore, the vector $\boldsymbol{v}_{ij}$ is the eigenvector of $\boldsymbol{S}_{ij}$ associated to the smallest eigenvalue. For the computation of the optical flow vector of each lattice point, we achieve appropriate decompositions to all lattice

points. This method is a mathematical formulation of the Lucus-Kanade method from view point of domain decomposition. The Lucus-Kanade method is an approximation of the variational energy-minimisation procedure with the piecewise constant assumption. For large n, the Lucus-Kanade method is an appropriate method and for small n, the Horn-Schunck and Nagel-Enkelmann methods are preferable methods for the computation of the detailed motion. Moreover, since the parameter α should select to satisfy the condition

$$\alpha = O(|\nabla f|). \tag{14}$$

Therefore, we can control the sequence of α_n as

$$\alpha_{n-1} \geq \alpha \tag{15}$$

since

$$|T_\sigma^n f| \leq |T_\sigma^n||f|, \ 0 < |T_\sigma| \leq 1. \tag{16}$$

For an appropriate decomposition of $\mathbf{D}_{ij}$, we can accept the Lucus-Kanade method as a relaxation method for the Horn-Schunck and Nagel-Enkelmann methods.

3 Smoothing Operation

We define a class of smoothing operation T_σ as

$$T_\sigma f(x,y) = \int\int_{\mathbf{R}^2} w(x-u, y-v; \sigma) f(u,v) du dv. \tag{17}$$

for a non-negative function $w(x,y)$ such that

$$\int\int_{\mathbf{R}^2} w(x,y) dx dy = 1. \tag{18}$$

The continuous version of the classical pyramid transform is achieved, setting

$$w(x,y) = \begin{cases} \frac{1}{\sigma^2}, & |x| \leq \sigma, \ |y| \leq \sigma, \\ 0, & \text{otherwise.} \end{cases} \tag{19}$$

The other traditional smoothing is the convolution with Gaussian

$$G(x,y) = \frac{1}{2\pi\sigma} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right). \tag{20}$$

For the sampled function $f_{ij} = f(i,j)$, eq. (17) is expressed as

$$T_\sigma f_{mn} = \sum_{m,n=\sigma} w_{i-m\, j-n} f_{mn}. \tag{21}$$

We write

$$T_\sigma f = f_{\sigma(1)}, \ T_\sigma^{n+1} f = T_\sigma(T_\sigma^n f) = f_{\sigma(n+1)}. \tag{22}$$

4 Dagstuhl Algorithms

The classical pyramid-transform-based method uses the optical flow $\boldsymbol{u}_n(t)$ computed from a pair $T_\sigma^n f(\boldsymbol{x}, t)$ and $T_\sigma^n f(\boldsymbol{x}, t+1)$ as the initial estimation of $\boldsymbol{u}_{n-1}(t)$ and refine $\boldsymbol{u}_{n-1}(t)$ from $T_\sigma^{n-1} f(\boldsymbol{x}, t)$ and $T_\sigma^{n-1} f(\boldsymbol{x} - \boldsymbol{u}_{n-1}(t), t+1)$. The operation

$$W_{\boldsymbol{d}} f(\boldsymbol{x}) = f(\boldsymbol{x} - \boldsymbol{d}) \tag{23}$$

is called warp, or the warp of f by the vector $\boldsymbol{d}$. Usually the vector $\boldsymbol{d}$ is shift-variant vector function.

On the other hand, we propose to compute $\boldsymbol{u}_{n-1}(t+1)$ from $T_\sigma^{n-1} f(\boldsymbol{x}, t+1)$ and $W(T_\sigma^{n-1} f(\boldsymbol{x} - \boldsymbol{u}_{n-1}(t), t+2))$.

Setting

$$f_k^n = T_\sigma^n f(\boldsymbol{x}, k), \ 0 \leq n \leq N, \tag{24}$$

the classical pyramid-based optical-flow computation algorithm is described in the following. In this algorithm, data are successive images

$$\begin{matrix} f_k^N & \cdots & f_{k+1}^N \\ \vdots & \cdots & \vdots \\ f_k^0 & \cdots & f_{k+1}^0 \end{matrix} \tag{25}$$

and $u_k^0 =: \boldsymbol{u}$, the operations are

$$u(f_k^l, f_{k+1}^l) = d_k^l \quad 0 \leq n \leq N$$

and

$$W(f_{k+1}^{l-1}, f_k^l) = f_{k+1}^{l-1}(\boldsymbol{x} - d_k^l, k+1)$$

Data: $f_k^N \cdots f_k^0$
Data: $f_{k+1}^N \cdots f_{k+1}^0$
Result: optical flow u_k^0
$n := N$;
while $n \neq 0$ **do**
  $u_k^n := u(f_k^n, f_{k+1}^n)$;
  $f_{k+1}^{n-1} := W(f_{k+1}^{n-1}, u_k^n)$;
  $d_k^{n-1} := u(f_k^{n-1}, f_{k+1}^{n-1})$;
  $\boldsymbol{u}_k^{n-1} := \boldsymbol{u}_k^n + \boldsymbol{d}_k^{n-1}$;
  $u_k^{n-1} := u_k^n + d_k^{n-1}$;
  $n := n - 1$
end

Algorithm 1: The Classical Pyramid Algorithm

The first dynamic algorithm computes optical flow using f_k^n and f_{k+1}^{n-1}. The second algorithm computes optical flow using f_k^n and f_{k+n}^{n-1}. In these algorithms, the resolutions of images depend on the frame number k.

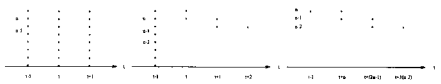
Data: $f_k^N \cdots f_k^0 \quad 1 \leq k \leq N$
Result: optical flow u_k^0
$n := N$;
$k := 0$;
while $n \neq 0$ **do**
  $d_k^n := u(f_k^n, f_{k+1}^n)$;
  $f_{k+1}^{n-1} := W(f_{k+1}^{n-1}, u_k^n)$;
  $d_k^{n-1} := u(f_k^{n-1}, f_{k+1}^{n-1})$;
  $u_k^{n-1} := u_k^n + d_k^{n-1}$;
  $n := n - 1$;
  $k := k - 1$
end

Algorithm 2: Dagstuhl Algorithm 1

Data: $f_k^N \cdots f_k^0 \quad 1 \leq k \leq M \quad M > N$
Result: optical flow u_k^0
$n := N$;
$k := 0$;
while $l \neq 0$ **do**
  $d_k^n := u(f_k^n, f_{k+l}^n)$;
  $f_{k+l}^{n-k} := W(f_{k+1}^{n-1}, u_k^n)$;
  $d_k^{n-1} := u(f_k^{n-1}, f_{k+1}^{n-1})$;
  $u_k^{n-1} := u_k^n + d_k^{n-1}$;
  $n := n - 1$
end

Algorithm 3: Dagstuhl Algorithm 2

Figure 1 shows the time-charst for three algorithms. The classical pyramid-based algorithm computes optical flow from all resolution images of fixed successive times as shown in (a). The Dagstuhl algorithm 1 propergates the flow vectors to the next successive frame pairs as shown in (b) The Dagstuhl algorithm 2 computes the flow vectors controlling the intervals between a pair of images as shown in (c).



(a) (b) (c)

Figure 1: Layer-Time Charts of Algorithms. (a) The classical pyramid-based algorithm computes optical flow from all resolution images of fixed successive times. (b)The Dagstuhl algorithm 1 propergates the flow vectors to the next successive frame pairs. (c)The Dagstuhl algorithm 2 computes the flow vectors controlling the intervals between pairs of images.

Figures 3, 4 and 5 show the results for these three algorithms for Marble Block sequence, Yosemite sequence, and our real image sequence captured by the camera mounted on a mobile robot. In these examples, N is three, the size of window is 5×5, and the algorithms extracted the flow vectors whose lengths are longer than 0.03. These results show the almost same results in appearances. We analyse mathematical properties of these three algorithms in the next section.

5 Mathematical Properties of Algorithms

The energy functional

$$E(\boldsymbol{u}, n, t; f) = \int\int_{\mathbf{R}^2} \{D(f_{\sigma(n)}, \boldsymbol{u}) + \alpha P(\boldsymbol{u})\}\, d\boldsymbol{x} \tag{26}$$

for

$$D(f, \boldsymbol{u}) = (\nabla f^\top \boldsymbol{u} + f_t), \; P(\boldsymbol{u}) = \nabla \boldsymbol{u}^\top \boldsymbol{A} \nabla \boldsymbol{u}, \tag{27}$$

is a convex if $\alpha \geq 0$. For this energy functional, we define

$$\boldsymbol{u}_n = \text{argument}\,(\min E(\boldsymbol{u}, n, t; f)). \tag{28}$$

Optical flow $\boldsymbol{u}_n$ is the solution of the equation

$$\nabla^\top \boldsymbol{A} \nabla \boldsymbol{u}_n = \frac{1}{\alpha} D(f_{\sigma(n)}, \boldsymbol{u}_n) \nabla f_{\sigma(n)}. \tag{29}$$

Multi-resolution optical-flow computation establishes an algorithm which guarantees the relation

$$\lim_{n \to 0} \boldsymbol{u}_n(t) = \boldsymbol{u}(t), \tag{30}$$

where $\boldsymbol{u}$ is the solution of

$$\nabla^\top \boldsymbol{A} \nabla \boldsymbol{u} = \frac{1}{\alpha} D(f, \boldsymbol{u}) \nabla f. \tag{31}$$

Since $E(\boldsymbol{u}, n, t; f)$ is a convex functional for a fixed f, this functional satisfies the relation

$$E(\boldsymbol{u}_n, n, t; f) < E(\boldsymbol{u}, n, t; f), \tag{32}$$

for $\boldsymbol{u}_n \neq \boldsymbol{u}$. Therefore, we have the relation

$$E(\boldsymbol{u}_{n-1}, n-1, t; f) \leq E(\boldsymbol{u}_n, n-1, t; f). \tag{33}$$

This relation implies that it is possible to generate a sequence which reaches to $E(\boldsymbol{u}_0, 0, t; f)$ from $E(\boldsymbol{u}_n, n-1, t; f)$, since

$$\begin{aligned} E(\boldsymbol{u}_{n-1}, n-1, t; f) &\leq E(\boldsymbol{u}_n, n-1, t; f) \\ E(\boldsymbol{u}_{n-1}, n-2, t; f) &\leq E(\boldsymbol{u}_n, n-2, t; f) \\ &\vdots \\ E(\boldsymbol{u}_1, 1, t; f) &\leq E(\boldsymbol{u}_2, 1, t; f) \\ E(\boldsymbol{u}_0, 0, t; f) &\leq E(\boldsymbol{u}_1, 0, t; f) \end{aligned} \tag{34}$$

for a fixed f, setting

$$\boldsymbol{u}_n = \boldsymbol{u}_{n-1} + \boldsymbol{d}_{n-1}, \tag{35}$$

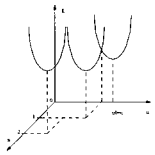


Figure 2: A Series of Convex Energy Functionals. The minimum of a fixed resolution derives an approximation of the minimum for the next fine-resolution for the optical flow computation

for a fixed time t where $|\boldsymbol{d}_{n-1}| \ll |\boldsymbol{u}_n|$. Figure 2 shows the sequence of minimums which derive approximations for the next fine-resolution images for the optical flow computation.

For the computation of $\boldsymbol{u}_{n-1}$, the algorithm minimises

$$E(\boldsymbol{d}_{n-1}, t; g) = \int\int_{\mathbf{R}^2} \left\{ (\nabla g^\top \boldsymbol{u} + g_t)^2 + P(\boldsymbol{d}_{n-1}) \right\} d\boldsymbol{x}, \tag{36}$$

for

$$\begin{aligned} g_(\boldsymbol{x} + \boldsymbol{d}_{n-1}, t+1) &:= W_{\boldsymbol{u}_n} f_{\sigma n-1}(\boldsymbol{x} - \boldsymbol{d}_{n-1}, t+1) \\ &= f_{\sigma(n-1)}(\boldsymbol{x} - \boldsymbol{u}_n, t+1) \quad (37) \\ g(\boldsymbol{x}, t) &:= f_{\sigma(n-1)}(\boldsymbol{x}, t), \quad (38) \end{aligned}$$

since $\boldsymbol{u}_n = \boldsymbol{u}_{n-1} + \boldsymbol{d}_{n-1}$.

It is also possible to use the solution of the parabolic equation

$$\frac{\partial \boldsymbol{u}_{n-1}}{\partial \tau} = \nabla^\top A \nabla \boldsymbol{u}_{n-1} + D(\boldsymbol{u}_{n-1}) D'(\boldsymbol{u}_{n-1}) \tag{39}$$

with the initial condition

$$\boldsymbol{u}_{n-1}(\boldsymbol{x}, 0) = \boldsymbol{u}_n. \tag{40}$$

We introduce the following definitions for the optical-flow vector $\boldsymbol{u}$.

Definition 1 *If the flow vectors satisfy the condition $|\boldsymbol{u}| \leq A$, we call the flow vector is A-stationary.*

Definition 2 *For sufficiently small positive constant β, if the optical-flow vector $\boldsymbol{u}$ satisfies the relation $\left|\frac{\partial \boldsymbol{u}}{\partial t}\right| \leq \beta$, we call the flow field is β-time stationary. Specially, if $\beta = 0$, the flow field is stationary in the temporal domain.*

Definition 3 *For sufficiently small positive constant γ, if the optical-flow vector $\boldsymbol{u}$ satisfies the relation $|\nabla \boldsymbol{u}| \leq \gamma$ in domain Ω, we call the flow field is γ-spatial stationary. Specially, if $\gamma = 0$, the flow field is stationary in the domain.*

Next, we introduce the cross-layer relation of flow vector.

Definition 4 *For*

$$\overline{\boldsymbol{u}_n} = \frac{1}{|\Omega|} \int_\Omega \boldsymbol{u}_n d\boldsymbol{x} \tag{41}$$

if

$$|\boldsymbol{u}_n - \overline{\boldsymbol{u}_n}| \leq \frac{\delta}{3}, \tag{42}$$

$$|\overline{\boldsymbol{u}_{n-1}} - \overline{\boldsymbol{u}_n}| \leq \frac{\delta}{3}, \tag{43}$$

$$|\boldsymbol{u}_{n-1} - \overline{\boldsymbol{u}_{n-1}}| \leq \frac{\delta}{3}, \tag{44}$$

we call $\boldsymbol{u}_n$ the δ-layer stationary.

Furthermore, setting

$$\Delta_n \boldsymbol{u}_n(\boldsymbol{x}, t) = \boldsymbol{u}_n(\boldsymbol{x}, t) - \boldsymbol{u}_{n-1}(\boldsymbol{x}, t), \tag{45}$$

we have the lemma.

Lemma 1 *If $\boldsymbol{u}_n$ is the δ-layer stationary, the relation $|\Delta_n \boldsymbol{u}_n(\boldsymbol{x}, t)| \leq \delta$ is satisfied.*

If $\boldsymbol{u}_{n-1}$ is the A-stationary, that is, $|\boldsymbol{u}_{n-1}| \leq \alpha$, we have the relation $|\overline{\boldsymbol{u}_n}| \leq \frac{\alpha}{\Omega}$. Since

$$|\boldsymbol{u}_{n-1} - \boldsymbol{u}_n| \leq |\boldsymbol{u}_{n-1} - \overline{\boldsymbol{u}_{n-1}}| + |\overline{\boldsymbol{u}_{n-1}} - \overline{\boldsymbol{u}_n}| + |\boldsymbol{u}_n - \overline{\boldsymbol{u}_n}|, \tag{46}$$

we have the relation

$$|\boldsymbol{u}_{n-1} - \boldsymbol{u}_n| \leq \delta \tag{47}$$

This relation leads to the next theorem.

Theorem 1 *If the optical flow vectors are δ-layer stationary, the traditional pyramid algorithm converges.*

Furthermore, since

$$\begin{aligned} &\boldsymbol{u}_{n-1}(t+1) - \boldsymbol{u}_n(t) \\ &= \boldsymbol{u}_{n-1}(t+1) - \boldsymbol{u}_n(t+1) + \boldsymbol{u}_n(t+1) - \boldsymbol{u}_n(t) \\ &= -\Delta_n \boldsymbol{u}_n(t+1) + \frac{\partial}{\partial t} \boldsymbol{u}_n, \end{aligned} \tag{48}$$

we have the relation

$$|\boldsymbol{u}_{n-1}(t+1) - \boldsymbol{u}_n(t)| \leq \delta + \left| \frac{\partial}{\partial t} \boldsymbol{u}_n \right| \leq \beta + \delta. \tag{49}$$

This relation leads to the next theorem.

Theorem 2 *Setting T_σ to be an image transform to derive a low resolution image from image f, the low resolution images are expressed as $T_\sigma f$. Here, we assume that T_σ stands for the shift-invariant operations. The regular pyramid transform, Low-pass filtering, and convolution with Gaussian are examples of this operator T_σ. For the pyramid transform and convolution with Gaussian, σ is the element of the non-negative integers $\mathbf{N}_+$ and non-negative real numbers $\mathbf{R}_+$, respectively. δ is sufficiently small and the motion is β-time stationary for sufficiently small constant, the algorithm 1 proposed in the previous section converge.*

Furthermore, since

$$\begin{aligned}&\boldsymbol{u}_n(\boldsymbol{x},t)-\boldsymbol{u}_{n-1}(\boldsymbol{x},t+k)\\&=\boldsymbol{u}_n(\boldsymbol{x},t)-\boldsymbol{u}_n(\boldsymbol{x},t+k)\\&+\boldsymbol{u}_n(\boldsymbol{x},t+k)-\boldsymbol{u}_{n-1}(\boldsymbol{x},t+k)\end{aligned}\tag{50}$$

and

$$\begin{aligned}&\boldsymbol{u}_n(\boldsymbol{x},t)-\boldsymbol{u}_n(\boldsymbol{x},t+k)\\&=\sum_{i=0}^{k-1}\{\boldsymbol{u}_n(\boldsymbol{x},t+i)-\boldsymbol{u}_n(\boldsymbol{x},t+(i+1))\},\end{aligned}\tag{51}$$

we have the relation

$$|\boldsymbol{u}_n(\boldsymbol{x},t)-\boldsymbol{u}_{n-1}(\boldsymbol{x},t+k)|\leq k\gamma+\delta.\tag{52}$$

This relation leads to the next theorem.

Theorem 3 *δ is sufficiently small and the motion is β-time stationary for sufficiently small constant, the algorithm 2 proposed in the previous section converge.*

Since the energy functional of the Lucus-Kanade method is convex, this energy functional generates a sequence of the solutions which converge the solution of the Lucus-Kanade method for the original images. Since physically

$$\min E(\boldsymbol{u},n,t;f)\leq\min E(\boldsymbol{u},n-1,t;f),\tag{53}$$

it is not easy to reach to $\boldsymbol{u}$ from $\boldsymbol{u}_N$ using minimisation strategy.

6 Discussion

If T_σ is a low-pass-filtering operation, the relation

$$T_\sigma^n f=T_\sigma f,\ n\geq 2\tag{54}$$

is satisfied. Therefore, we have the relation

$$\min E(\cdot,n,t;f)=\min E(\cdot,1,t,f).\tag{55}$$

This relation means that two layers are sufficient for multi-resolution analysis. Furthermore, for the achievement of multi-resolution analysis, the smoothing operation must satisfies the relation

$$T_\sigma^m f\neq T_\sigma^n\neq f,\ m\neq n.\tag{56}$$

If an image is a collection of randomly moving points, this images is expressed as

$$f(x,y,t)=\sum_{i=1}^{n}\mathbf{1}(\boldsymbol{x}-\boldsymbol{x}_i(t))\ \boldsymbol{x}_i=(x_i,y_i)^\top,\tag{57}$$

for randomly moving point $\{\boldsymbol{x}_i\}_{i=1}^n$. Generally this image does not satisfy the convergence condition derived in this paper.

The Lucus-Kanade method is an approximation of the variational energy-minimisation procedure with the piecewise constant assumption. For large n, the Lucus-Kanade method is an appropriate method and for small n, the Horn-Schunck and Nagel-Enkelmann methods are preferable methods for the computation of the detailed motion. Moreover, since the parameter α should select to satisfy the condition $\alpha=O(|\nabla f|)$. Therefore, we can control the sequence of α_n as $\alpha_{n-1}\geq\alpha_n$, since

$$|T_\sigma^n f|\leq|T_\sigma^n||f|,\ 0<|T_\sigma|\leq 1.\tag{58}$$

For the displacement vector $\boldsymbol{u}$, we define the functional

$$J(\boldsymbol{u})=\int\int_{\mathbf{R}^2}|f(\boldsymbol{x}-\boldsymbol{u},t+1)-f(\boldsymbol{x},t)|^2.\tag{59}$$

Since

$$J(\boldsymbol{u})=2E-2C(\boldsymbol{u}),\tag{60}$$

where

$$\begin{aligned}E&=\int\int_{\mathbf{R}^2}|f(\boldsymbol{x}-\boldsymbol{u},t+1)|^2d\boldsymbol{x}\\&=\int\int_{\mathbf{R}^2}|f(\boldsymbol{x},t)|^2d\boldsymbol{x},\end{aligned}\tag{61}$$

$$C(\boldsymbol{u})=\int\int_{\mathbf{R}^2}f(\boldsymbol{x}-\boldsymbol{u},t+1)f(\boldsymbol{x},t)^2d\boldsymbol{x},\tag{62}$$

minimisation of $J(\boldsymbol{u})$ is achieved by maximisation of $C(\boldsymbol{u})$. The functionals $J(\boldsymbol{u})$ and $C(\boldsymbol{u})$ are convex and concave functionals, respectively. Therefore, we can apply the multi-resolution method to minimise $J(\boldsymbol{u})$. Moreover, if the motion satisfies appropriate conditions similar to these derived in the previous section, the algorithm generates the converging sequence.

Same as warping in the algorithm, for the computation of $\boldsymbol{d}_{n-1}$, we compute

$$\boldsymbol{d}_{n-1}=\text{argument}(\max C_{n-1}(\boldsymbol{d})),\tag{63}$$

where

$$C_{n-1}=\int\int_{\mathbf{R}^2}f_{\sigma(n-1)}(\boldsymbol{x},t)W_{\boldsymbol{u}_n}f(\boldsymbol{x}-\boldsymbol{d},t+1)d\boldsymbol{x}.\tag{64}$$

In the correlation-based method, the maximum of the $C(\boldsymbol{u})$ is searched by decreasing the size of the window. This is a multi-resolution strategy. In this paper, in stead of controlling the size of the windows, we propose controlling of the resolution of images [5].

The multi-resolution analysis in this paper is valid for feature extraction from images, if the feature for extraction is the solution of a convex optimisation problem. Therefore, we have the next assertion.

Assertion 1 *If a feature for pattern analysis is the solution of a unimodal optimisation problem, it is possible to construct a multi-resolution algorithm which guarantees the convergence on a sequence of solution.*

Furthermore, if the energy functional of a discrete problem satisfies unimodality, it is possible to computer the minimum as the solution using appropriate discrete optimisation technique [6]. For instance, graph-cut solves optimisation problems in early vision expressed as attribute graph labelling [6].

7 Conclusions

We have introduced a class of multi-resolution optical flow computation algorithm as extensions of the Lucus-Kanade method with the pyramid transform. These algorithms are a combination of variational method and multi-resolution analysis of image. We have shown the convergence property of the algorithms for a condition.

In this paper, we accepted that the Lucus-Kanade method is a relaxation for the variational energy minimisation. However, the approximation ratio of Lucus-Kanade method to the variational energy minimisation is not mathematically shown.

References

[1] Ruhnau, P., Knhlberger, T., Schnoerr, C., Nobach, H. Variatinal optical flow estimation for particle image velocimetry, Experiments in Fluids, **38**, 21-32, 2005.

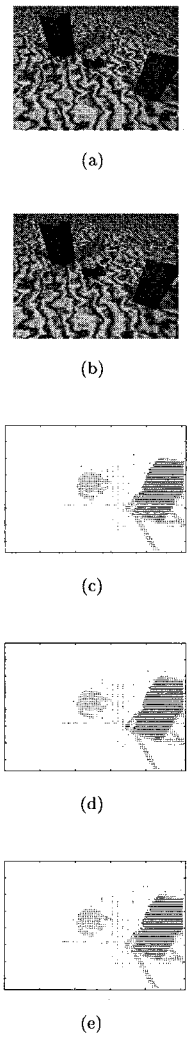
[2] Horn, B. K. P. and Schunck, B. G., Determining optical flow, Artificial Intelligence, **17**, 185-204, 1981.

[3] Nagel, H.-H., On the estimation of optical flow:Relations between different approaches and some new results, Artificial Intelligence, **33**, 299-324, 1987.

[4] Barron, J. L., Fleet, D. J., Beauchemin, S. S., Performance of optical flow techniques, International Journal of Computer Vision, **12**, 43-77, 1995.

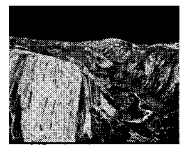
[5] Beck, C., Bayerl, P., Neumann, H., Optic flow integration at multiple spatial frequencies - Neural mechanism and algorithm, LNCS **4291**, 741-750, 2006.

[6] Freedman, D., Drineas, Energy minimization via graph cuts: setting what is possible. CVPR 2005, 939-946, 2005.

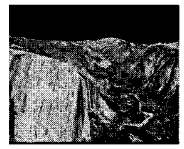


(a) (b) (c) (d) (e)

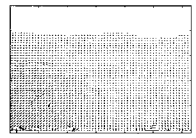
Figure 3: Results (a), (b) Image Sequence Marble Block. (c) The Classical Pyramid Method. (c) Algorithm 1. (e) Algorithm 2.



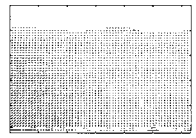
(a)



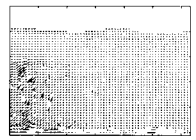
(b)



(c)

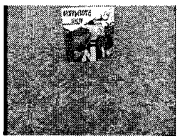


(d)



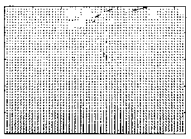
(e)

Figure 4: Results (a), (b) Image Sequence Yosemite. (c) The Classical Pyramid Method. (d) Algorithm 1. (e) Algorithm 2.

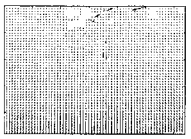


(a)

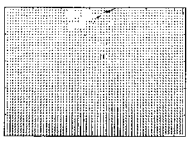
(b)



(c)



(d)



(e)

Figure 5: Results. Real Image Sequence captured by a Robot. (c) The Classical Pyramid Method. (d) Algorithm 1. (e) Algorithm 2.