Unit gradient vectors based motion estimation techniques

Pramuk Boonsieng[1], **Toshiaki Kondo**[2], and **Waree Kongprawechnon**[3], Non-members

ABSTRACT

This paper presents two motion estimation techniques based on gradient orientation information (GOI). We utilize GOI by means of unit gradient vectors (UGVs) of an image. UGVs are used, in place of image intensities, in two conventional motion estimation techniques: the gradient method (GM) and the gradient structure tensor method (GSTM). We name the two novel approaches the gradient orientation based gradient method (GOGM) and gradient orientation structure tensor method (GOSTM), respectively. The proposed methods have been compared with the traditional methods, the GM and GSTM, under mildly noisy (40dB) and uniformly varying lighting conditions. As the gradient orientation is remarkably invariant to uniform changes of image intensities, simulation results show that the proposed methods can perform motion estimation robustly in changing lighting conditions. The results also reveal the significance of a low-pass filter to be applied to an input image sequence prior to the computation of motion vectors. The paper also discusses the appropriate size of the local area where a motion vector is estimated.

Keywords: Motion Estimation, Motion Vectors, Gradient Method, Gradient Structure Tensor Method, Unit Gradient Vectors.

1. INTRODUCTION

Motion estimation is an important task in the fields of image sequence processing and computer vision. It finds various applications, including image stabilization for digital cameras and camcorders, surveillance systems, image registration, video coding, dynamic scene analysis, object detection and tracking etc. Motion estimation techniques, especially in the spatial domain, may be classified into two categories, gradient-based methods [1], [9] and correlation-based methods [2], [4](commonly referred to as block matching or template matching). The former technique is often used to obtain a dense optical flow field or motion vectors. The technique is, however, effective only when the displacement between images over time is small, typically, a few pixels only. Meanwhile, the latter technique is the most intuitive approach to compute motion vectors in that it searches for a similar pattern between two images. Since the latter category of methods can handle larger displacements, it is used not only for motion estimation, but also for establishing correspondence between images captures at different viewpoints such as stereopsis and image mosaicing. It is, however, not suitable for computing dense motion vectors owing to its high computational cost.

This paper is concerned with the former approach, i.e., gradient-based methods. We propose to use the unit gradient vectors (UGVs) in place of image intensities, for the two conventional motion estimation methods, the gradient method (GM) [1], [5], [17] and the gradient structure tensor method (GSTM) [6]-[8]. We name the two proposed methods the gradient orientation-based gradient method (GOGM) and the gradient orientation structure tensor method (GOSTM), respectively. Simulation results show that the proposed methods work robustly under changing lighting conditions. In addition, the paper describes the significance of the low-pass filter to be applied to input image sequences prior to the computation of motion vectors. Furthermore, the paper presents the effect of the size of a local area (so-called a block) where a motion vector is estimated. The results show that the appropriate selection of a low-pass filter and the size of the block are significant factors for all the gradient-based methods to improve their motion estimation performances.

2. GRADIENT-BASED MOTION ESTIMATION METHODS

Gradient-based methods exploit the relationship between the spatial and temporal gradients of intensity. This relationship can be used to segment images based on the velocity of points [9].

2.1 Gradient Method

The gradient method (GM) is more officially referred to as the spatio-temporal gradient method in the literature, but we call it the GM in this paper. When we assume that image intensity is constant over time, we have

Manuscript received on May 6, 2009 ; revised on May 24, 2010.

[1,2,3] The authors are with Sirindhorn International Institute of Technology, Thammasat University - Rangsit Campus P.O.Box 22, Pathum Thani 12121, Tel. +66 (0) 2986 9009, 2986 9101, Fax. +66(0) 2986 9112-3. , Thailand. E-mail: b.pramuk@gmail.com, tkondo@siit.tu.ac.th and waree@siit.tu.ac.th

$$g(x, y, t) = g(x + dx, y + dy, t + dt) \tag{1}$$

where $g(x, y, t)$ indicate the image intensities at points (x, y) at time t, and (dx, dy) denote the image displacements over time dt.

By taking the first-order Taylor series expansion of the right hand side of Eq. (1), we obtain

$$g(x+dx, y+dy, t+dt) \approx g(x, y, t)+g_x \cdot dx+g_y \cdot dy+g_t \cdot dt \tag{2}$$

where the subscripts denote partial derivatives. From Eqs. (1) and (2), one can derive the well-known optical flow constraint equation (OFCE)

$$g_x \cdot u + g_y \cdot v + g_t = 0 \tag{3}$$

where $(u, v) = (dx/dt, dy/dt)$ denote the optic flow, and $(g_x, g_y, g_t) = (\partial g(x, y, t)/\partial x, \partial g(x, y, t)/\partial y, \partial g(x, y, t)/\partial t)$ express the spatio-temporal image gradients. Eq. (3) is apparently underdetermined because it contains two unknowns, u and v. To solve the equation for the optical flow (u, v), we may introduce either a local or global smoothness constrain. The former constraint assumes that the motion is constant within the small area where a motion vector is estimated [10], [11], while the latter assumes that the optical flow changes smoothly over the entire image [1], [9]. A comparative study between these two optimization techniques can be found in Ref. [10]. The local approach is more suitable for the applications that require well-defined boundaries between areas with different motions because each motion vectors is computed within a block independently of its neighboring blocks. On the contrary, the boundaries by the global approach are blurred because of its smoothing operation among adjacent motion vectors. Under the local smoothness constraint, the optical flow (u, v) can be determined by minimizing the quadratic cost-function, F

$$F = \sum \left(g_x \cdot u + g_y \cdot v + g_t\right)^2 \tag{4}$$

where $\sum$ indicates the summation over a small region or a block for computing each flow vector. The least squares solution of Eq. (4) is given by

$$\begin{cases} u = \dfrac{\left(\sum g_x g_y\right)\left(\sum g_y g_t\right) - \left(\sum {g_y}^2\right)\left(\sum g_x g_t\right)}{\left(\sum {g_x}^2\right)\left(\sum {g_y}^2\right) - \left(\sum g_x g_y\right)^2} \\ v = \dfrac{\left(\sum g_x g_y\right)\left(\sum g_x g_t\right) - \left(\sum {g_x}^2\right)\left(\sum g_y g_t\right)}{\left(\sum {g_x}^2\right)\left(\sum {g_y}^2\right) - \left(\sum g_x g_y\right)^2} \end{cases}. \tag{5}$$

2.2 Gradient Structure Tensor Method

Following the same notation as above, the 3-D gradient structure tensor (GST) is expressed by

$$GST = \begin{pmatrix} \sum {g_x}^2 & \sum g_x g_y & \sum g_x g_t \\ \sum g_y g_x & \sum {g_y}^2 & \sum g_y g_t \\ \sum g_t g_x & \sum g_t g_y & \sum {g_t}^2 \end{pmatrix} \tag{6}$$

where $\sum$ here denotes the integration within a small 3-D space such as a cube. g_x, g_y, and g_t denote the partial derivatives of image intensities g along the x, y and t axes. To obtain these partial derivatives, we use a 3-D version of the Sobel operators [12]. The three 2-D masks in Fig. 1 form a cubic of 3×3×3 voxels and serves as a 3-D filtering kernel. We can compute g_x, g_y, and g_t by appropriately rotating the 3-D mask.

1	2	1
0	0	0
-1	-2	-1

2	3	2
0	0	0
-2	-3	-2

1	2	1
0	0	0
-1	-2	-1

Fig. 1: *3-D Sobel operators that form a cube of* $3 \times 3 \times 3$ *voxels.*

The 3-D GST can be viewed as a correlation matrix of the spatio-temporal gradient vectors in principal component analysis (PCA) [7]. Hence, motion estimation is formulated as a total squares problem or orthogonal regression in 3-D space. In this case, the third eigenvector $V_3 = [x_3 \quad y_3 \quad t_3]^T$, associated with the smallest eigenvalue λ_3 ($\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$) of the GST, indicates the motion trajectory in spatio-temporal space (x, y, t). We may then obtain optical flow (u, v) as

$$\begin{cases} u = x_3/t_3 \\ v = y_3/t_3 \end{cases}. \tag{7}$$

It is relevant to note that the confidence of each motion vector obtained by the method can be evaluated by analyzing eigenvalues of the structure tensor [11]. When the three eigenvalues have the relationship as $\lambda_1 \approx \lambda_2 \gg \lambda_3 \geq 0$, the motion trajectory form a linear structure in spatio-temporal space, which indicates a high confidence of the estimated motion. When the relationship $\lambda_1 \gg \lambda_2 \approx \lambda_3 \geq 0$ holds, the motion trajectory is plane-like, which indicates the aperture problem where we can estimate motion only along the gradient vector available. When the relationship $\lambda_1 \approx \lambda_2 \approx \lambda_3 > 0$ holds, the corresponding local region may have impulsive noise or motion discontinuities. Finally, if $\lambda_1 \approx \lambda_2 \approx \lambda_3 \approx 0$ holds, there is no gradient information and motion estimation is made impossible. The gradient structure tensor method (GSTM) is a new alternative approach to the GM and has been actively studied in recent years [3], [13], [14].

3. PROPOSED METHODS

3.1 Unit Gradient Vectors

All the motion estimation techniques based on the OFCE are susceptible to the changes of image intensities often caused by varying lighting conditions. We propose to use gradient orientation information (GOI) in place of conventional image features such as intensities and gradients, because gradient orientation is known to be insensitive under varying illumination [15], [16].

We describe how to extract gradient orientation information (GOI) of an image. Let $g(x, y)$ be the image intensities at pixel coordinates (x, y). The gradient vectors of $g(x, y)$ are approximated by partial derivatives

$$\begin{cases} g_x(x,y) = g(x,y) * k_x \\ g_y(x,y) = g(x,y) * k_y \end{cases} \tag{8}$$

where the symbol $*$ denotes convolution, and k_x and k_y are first-derivative operators in the x and y directions, respectively. The Sobel operators may be used as the first-derivative operators. GOI can be obtained as unit gradient vectors (UGVs) that are computed by dividing gradient vectors by their magnitudes:

$$\begin{cases} n_x(x,y) = g_x(x,y) \Big/ \sqrt{{g_x}^2(x,y) + {g_y}^2(x,y)} \\ n_y(x,y) = g_y(x,y) \Big/ \sqrt{{g_x}^2(x,y) + {g_y}^2(x,y)} \end{cases} \tag{9}$$

where we assign zeros to (n_x, n_y) if the denominator is equal to zero. In this way, we extract UGVs by discarding the magnitude information of the gradient vectors. The n_x and n_y patterns of the Lena image are shown in Fig. 2. As can be seen, horizontal lines, i.e., vertical gradients are rich in n_x (Fig. 2(b)), while vertical lines, i.e., horizontal gradients are dominant in n_y (Fig. 2(c)).

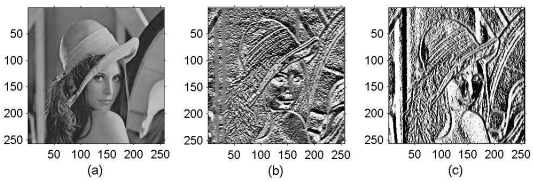


Fig.2: *(a) An 8-bit gray-scale image, (b) an unit gradient pattern in x direction, n_x, (c) an unit gradient pattern in y direction, n_y.*

3.2 Gradient Orientation Based Gradient Method

We devise a new technique for motion estimation by introducing UGVs to the GM, which we call the gradient orientation based gradient method or GOGM. We make use of the UGVs, n_x and n_y, in place of image gradient, g, as expressed in Eq. (10). This is possible because n_x and n_y are scalars as g is

$$\begin{cases} n_x(x,y,t) = n_x(x+dx, y+dy, t+dt) \\ n_y(x,y,t) = n_y(x+dx, y+dy, t+dt) \end{cases} \tag{10}$$

where n_x and n_y indicate the UGVs in x and y directions, respectively. And by taking the first-order Taylor series expansion of the right hand side, we can get the new OFCEs as shown in Eq. (11)

$$\begin{cases} n_{xx} \cdot u_1 + n_{xy} \cdot v_1 + n_{xt} = 0 \\ n_{yx} \cdot u_2 + n_{yy} \cdot v_2 + n_{yt} = 0 \end{cases} \tag{11}$$

where the first subscript of each element denotes which of x and y components of the UGVs are used, and the second the direction of the partial derivative, either x, y, or t. Thus, we can obtain two estimates of image motion (u_1, v_1) and (u_2, v_2) as expressed in Eqs. (12) and (13) for an unit gradient pattern in x and y direction, respectively.

$$\begin{cases} u_1 = \dfrac{(\sum n_{xx} n_{xy})(\sum n_{xy} n_{xt}) - (\sum {n_{xy}}^2)(\sum n_{xx} n_{xt})}{(\sum {n_{xx}}^2)(\sum {n_{xy}}^2) - (\sum n_{xx} n_{xy})^2} \\ v_1 = \dfrac{(\sum n_{xx} n_{xy})(\sum n_{xx} n_{xt}) - (\sum {n_{xx}}^2)(\sum n_{xy} n_{xt})}{(\sum {n_{xx}}^2)(\sum {n_{xy}}^2) - (\sum n_{xx} n_{xy})^2} \end{cases} \tag{12}$$

$$\begin{cases} u_2 = \dfrac{(\sum n_{yx} n_{yy})(\sum n_{yy} n_{yt}) - (\sum {n_{yy}}^2)(\sum n_{yx} n_{yt})}{(\sum {n_{yx}}^2)(\sum {n_{yy}}^2) - (\sum n_{yx} n_{yy})^2} \\ v_2 = \dfrac{(\sum n_{yx} n_{yy})(\sum n_{yx} n_{yt}) - (\sum {n_{yx}}^2)(\sum n_{yy} n_{yt})}{(\sum {n_{yx}}^2)(\sum {n_{yy}}^2) - (\sum n_{yx} n_{yy})^2} \end{cases} \tag{13}$$

3.3 Gradient Orientation Structure Tensors

In the gradient structure tensor method (GOSTM), image sequences, i.e., spatio-temporal images are processed in the 3-D space spanned by x, y and t axes. Firstly, the 2-D Sobel operators are applied on every frames in image sequences and, then, n_x and n_y can be calculated by dividing gradient orientations by their magnitude. According to 3-D *GST* that is expressed by Eq. (6), we propose to utilize the unit gradient vectors n_x and n_y in place of image gradient g. Therefore, we have two tensors as expressed in Eqs. (14) and (15),

$$GOST_{n_x} = \begin{pmatrix} \sum {n_{xx}}^2 & \sum n_{xx} n_{xy} & \sum n_{xx} n_{xt} \\ \sum n_{xy} n_{xx} & \sum {n_{xy}}^2 & \sum n_{xy} n_{xt} \\ \sum n_{xt} n_{xx} & \sum n_{xt} n_{xy} & \sum {n_{xt}}^2 \end{pmatrix} \tag{14}$$

and

$$GOST_{n_y} = \begin{pmatrix} \sum {n_{yx}}^2 & \sum n_{yx} n_{yy} & \sum n_{yx} n_{yt} \\ \sum n_{yy} n_{yx} & \sum {n_{yy}}^2 & \sum n_{yy} n_{yt} \\ \sum n_{yt} n_{yx} & \sum n_{yt} n_{yy} & \sum {n_{yt}}^2 \end{pmatrix} \tag{15}$$

where the first subscript of each element denotes which of x and y components of the unit gradient vectors are used, and the second the direction of the partial derivative, either x, y, or t. The former is obtained using n_x, i.e., the x components of the unit gradient vectors, while the latter is computed using n_y, the y components of the unit gradient vectors.

3.4 Confidence Measurement

Since, we have n_x and n_y for the unit gradient vectors in x and y directions, we may obtain two motion estimates (u_1, v_1) and (u_2, v_2), by applying PCA to GOSTs.

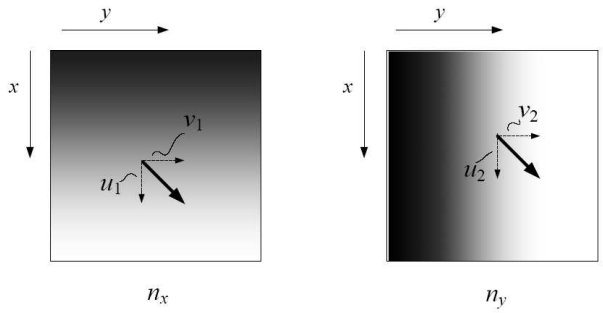


Fig.3: *The illustrations (a) and (b) show typical patterns of n_x and n_y in which the gradient tends to be one directional.*

Fig. 3 illustrates the typical patterns of unit gradient vectors in x direction, n_x, and in y direction, n_y. An unit gradient vectors are processed in the 2-D space spanned by x and y axes. The vectors u and v are the estimated motion vectors in x and y axes, respectively. The subscripts 1 and 2 are used for defining the estimated motion vector in pattern n_x and n_y, respectively. The two motion estimates are supposed to be identical, but never be the same in practice. Hence, these two estimates need to be combined by using some weighting factor for each motion estimate. The covariance matrix of the two patterns n_x and n_y is expressed by

$$N = \begin{pmatrix} \sum {n_x}^2 & \sum n_x n_y \\ \sum n_y n_x & \sum {n_y}^2 \end{pmatrix} \tag{16}$$

where $\sum$ indicates the integration area in a local 2-D space. By applying PCA to this matrix, we obtain two eigenvalues λ_1 and λ_2 where $\lambda_1 > \lambda_2 = 0$. We use these eigenvalues as weights as in Eq. (17).

$$\begin{cases} u = (\lambda_1/(\lambda_1+\lambda_2)) \cdot u_1 + (\lambda_2/(\lambda_1+\lambda_2)) \cdot u_2 \\ v = (\lambda_2/(\lambda_1+\lambda_2)) \cdot v_1 + (\lambda_1/(\lambda_1+\lambda_2)) \cdot v_2 \end{cases} \tag{17}$$

When the gradients are one-directional, the two eigenvectors satisfy the condition $\lambda_1 \gg \lambda_2 \approx 0$. This is the case that the aperture problem occurs, and we can obtain reliable motion estimation only in the direction along the gradients. When the gradients are multi-directional, holding the relationship $\lambda_1 \approx \lambda_2 \gg 0$, two motion estimates will be equally reliable. In this way, the weights are automatically adjusted according to the diversity of gradients.

4. COMPARATIVE STUDY ON THE FOUR APPROACHES

For the quantitative evaluation of the four motion estimation techniques, GM, GOGM, GSTM, and GOSTM, a synthetic image sequence is created by shifting an 8-bit gray-scale image of size 256 by 256 pixels by a known amount of pixels. We generate five synthetic image sequences with five test images that are widely used in the image processing community, i.e., Lena, Girl, Peppers, House, and Mandrill images. For example, Fig. 4 shows three consecutive frames using the House image where the image is shifted from one frame to the next by 2 pixels in both horizontal (rightward) and vertical (downward) directions. The GM and GOGM need only two frames for estimating motions, while the GSTM and GOSTM require three frames.

Fig.4: *A synthetic image sequence with the motion of (2,2) pixels generated with the House image. (a) Previous, (b) current, and (c) next frames.*

We assume that motion estimation is successful when the following condition is satisfied:

$$MAX(|\Delta x|, |\Delta y|) \leq 0.5 \tag{18}$$

where Δx and Δy denote motion estimation errors in pixels units in x and y directions, respectively. The condition indicates that estimated motion vectors must point correct pixels to be assumed successful.

It is a necessary step for all gradient-based methods to compute gradients of an image, g_x and g_y. We use the Sobel operators as first-derivative operators because of its reasonably good performance and ease of use. The difference between two successive frames is used as g_t. On the other hand, for the GSTM and GOSTM, we use a 3-D version of the Sobel operators [12]. The three masks in Fig. 1 form a cubic of 3 × 3 × 3 voxels and serves as a 3-D convolution mask. The orientation of the 3-D mask must be selected properly to compute g_x, g_y, and g_t, respectively.

4.1 Characteristics of the Methods

We compare the performances of the GM, GOGM, GSTM and GOSTM in a condition where images intensities are constant over time under a mildly noisy condition (SNR=40dB). Zero-mean Gaussian noise is added to each image sequence to generate a noisy image sequence. We have first tested the impact of the size of the local area for summation, i.e., the size of block in Eqs. (5) and (6). We have tested two sizes of 8×8 and 16×16 pixels for the GM and GOGM and $8 \times 8 \times 3$ and $16 \times 16 \times 3$ voxels for the GSTM and GOSTM. We choose these two sizes because they are wildly used in the video compression applications. When the size of block is 8×8 pixels, we have 28×28 motion vectors in the entire image, whereas we have 14×14 motion vectors when block is set 16×16 pixels. We observe the success rates on the test images for comparison as shown in Tables 1-5. It is evident that a larger size of local area helps to improve the motion estimation performance for all the four techniques.

Table 1: *Successful motion estimation rate in percentage(%) for the Lena image when the true motion is (2,2) with SNR 40 dB.*

Methods		GM		GOGM		GSTM		GOSTM	
Size of blocks		8x8	16x16	8x8	16x16	8x8	16x16	8x8	16x16
Size of LPF mask	3x3	4.97	9.18	1.79	0.00	46.68	56.63	36.10	43.37
	5x5	9.95	20.92	12.50	14.29	60.59	78.57	61.48	79.59
	7x7	14.80	25.51	20.92	44.39	70.03	91.84	78.19	92.86
	9x9	14.41	30.10	26.40	54.59	83.16	92.86	88.27	95.92
	11x11	18.37	37.24	28.70	61.73	87.63	96.94	91.45	97.45
	13x13	18.11	39.80	29.34	62.76	91.07	98.98	92.47	97.96
	15x15	22.19	39.80	29.85	58.67	95.66	98.98	90.69	93.88

Table 2: *Successful motion estimation rate in percentage (%) for the Girl image when the true motion is (2,2) with SNR 40 dB.*

Methods		GM		GOGM		GSTM		GOSTM	
Size of blocks		8x8	16x16	8x8	16x16	8x8	16x16	8x8	16x16
Size of LPF mask	3x3	10.71	17.35	3.19	0.51	70.92	87.24	47.96	65.31
	5x5	18.24	37.24	17.09	25.51	80.74	93.88	76.02	89.29
	7x7	20.66	47.96	28.70	63.27	87.12	96.43	86.99	93.37
	9x9	22.45	50.51	34.44	68.88	91.96	94.90	90.18	95.92
	11x11	23.85	52.04	35.46	73.47	93.24	98.47	90.31	95.92
	13x13	24.62	56.63	40.94	75.00	95.66	99.49	92.73	91.84
	15x15	27.81	54.59	41.58	72.96	96.17	98.98	90.18	94.39

Table 3: *Successful motion estimation rate in percentage (%) for the Pepper image when the true motion is (2,2) with SNR 40 dB.*

Methods		GM		GOGM		GSTM		GOSTM	
Size of blocks		8x8	16x16	8x8	16x16	8x8	16x16	8x8	16x16
Size of LPF mask	3x3	10.59	15.82	4.46	0.51	65.94	80.61	56.25	73.47
	5x5	14.92	36.73	20.92	31.63	77.04	90.82	82.40	96.43
	7x7	20.28	44.90	30.36	62.24	85.97	95.41	93.62	100.00
	9x9	22.19	52.04	32.14	71.94	89.29	96.43	97.19	98.98
	11x11	24.36	50.00	33.80	74.49	92.98	97.96	98.47	100.00
	13x13	27.55	52.04	38.39	74.49	94.13	98.98	96.94	98.47
	15x15	28.32	53.06	40.43	71.43	96.05	99.49	94.13	98.98

Table 4: *Successful motion estimation rate in percentage (%) for the House image when the true motion is (2,2) with SNR 40 dB.*

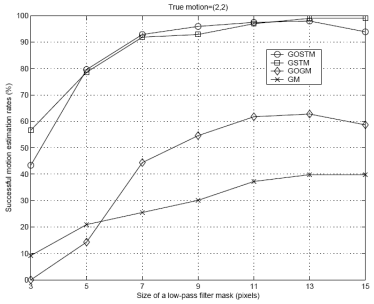
Methods		GM		GOGM		GSTM		GOSTM	
Size of blocks		8x8	16x16	8x8	16x16	8x8	16x16	8x8	16x16
Size of LPF mask	3x3	2.68	5.61	0.89	0.00	36.10	48.98	22.96	25.00
	5x5	7.02	17.86	8.42	7.65	46.17	65.31	41.20	60.20
	7x7	7.27	22.96	18.75	23.47	54.46	72.96	57.27	68.37
	9x9	10.20	20.92	19.26	41.84	60.84	72.96	61.35	75.00
	11x11	11.86	28.06	22.70	47.96	64.67	77.04	63.27	70.92
	13x13	14.67	26.53	23.85	42.35	66.96	82.14	60.97	69.90
	15x15	14.54	30.61	28.83	45.92	71.81	77.04	64.29	69.39

Table 5: *Successful motion estimation rate in percentage (%) for the Mandril image when the true motion is (2,2) with SNR 40 dB.*

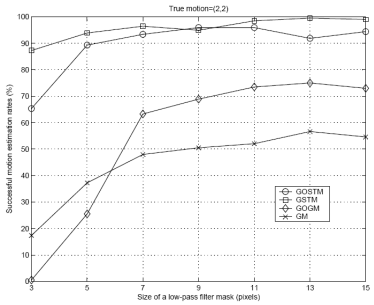
Methods		GM		GOGM		GSTM		GOSTM	
Size of blocks		8x8	16x16	8x8	16x16	8x8	16x16	8x8	16x16
Size of LPF mask	3x3	3.02	1.65	0.41	0.00	57.97	67.58	34.89	41.76
	5x5	9.07	10.44	10.30	5.49	71.43	89.01	68.82	85.71
	7x7	14.15	26.92	23.49	39.56	83.65	96.15	92.72	98.90
	9x9	17.03	33.52	31.32	64.29	93.68	97.80	97.25	100.00
	11x11	18.13	39.56	33.65	71.98	96.15	99.45	99.31	99.45
	13x13	21.15	46.15	36.54	75.27	96.70	99.45	98.76	99.45
	15x15	23.63	47.80	39.29	68.68	97.53	99.45	94.51	97.80

It is known that gradient-based methods are sensitive to high frequency components in an image and an application of a certain low-pass filter is always necessary prior to motion estimation. Thus, we have next tested the accuracies of the four motion estimation methods when the size of the low-pass filter mask is varied from 3×3 to 15×15 pixels. The low-pass filter used here is a simple Gaussian filter. The size of block is set 16×16 pixels in this part. We observe the successful motion estimation rates as plotted in Fig. 5 where the abscissa denotes the size of the low-pass filter mask and the ordinate indicates the successful motion estimation rates. As shown in Fig. 5, larger size of low-pass filters help to achieve higher motion estimation success rates. It should be noted, however, that image details will be lost more when a larger size of low-pass filter is used. From Fig. 5, we can state that low-pass filters of size 11×11 or 13×13 pixels may be a good compromise between the accuracy and resolution of estimated motion vectors.

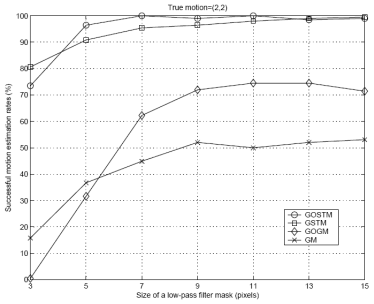
Under mildly noisy (40dB) and constant lighting conditions, the GSTM and GOSTM work equally well. There are noticeable gaps in the successful motion estimation rates between the GM and the other three methods, about half of the other three techniques, regardless of the size of the low-pass filter and block used. The success rate of the GOGM is approximately the two-third of that of the GSTM and GOSTM for the low-pass filter of size 11×11 and 13×13 pixels. Fig. 6 shows part of the motion vectors estimated by the four techniques. In Figs. 6(a) and 6(c), most of the motion vectors tend to have large motion estimation errors in the direction tangent to the edges of the door, the window, and the gutter. The faulty motion vectors are due to the aperture



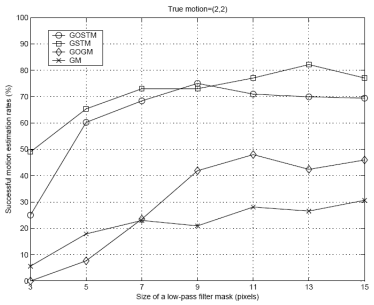
(a)



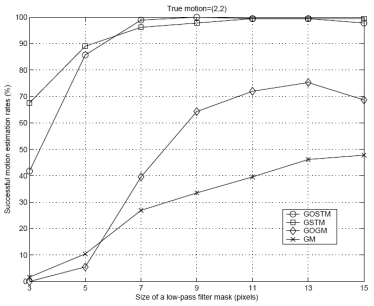
(b)



(c)



(d)

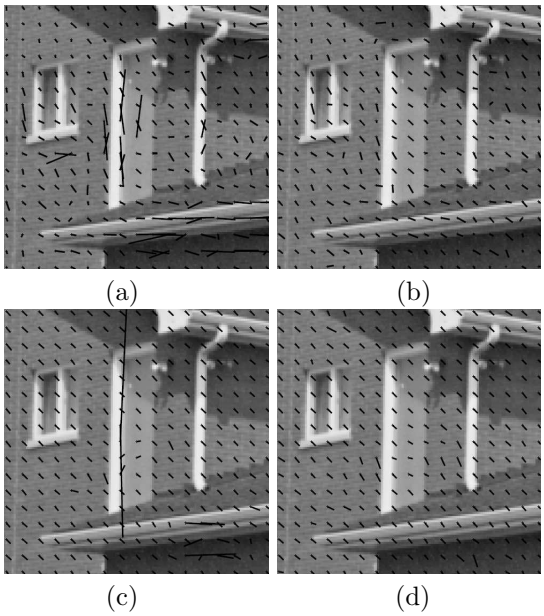


(e)

Fig.5: *Success rates for the (a) Lena, (b) Girl, (c) Pepper, (d) House, and (e) Mandril images when images intensities are constant over time and a mildly noisy condition (40dB) is added. Block size is set* 16×16 *pixels.*

problem where only one directional gradient is available in a local area. Many of these failed motion estimations have been improved by the proposed method (GOGM and GOSTM) as shown in Figs. 6(b) and 6(d). This is achieved in the combining step of two motion estimates in which the weighting factors are adapted depending on the variety of the gradient orientation in each local area. Consequently, large erroneous motion estimates are successfully suppressed. Furthermore, Fig. 4.1 shows the weighting factor, $\lambda_2/(\lambda_1 + \lambda_2)$, as a confidence analysis of estimated motions. The weighting factor is scaled from 0 to 255 for visualization purposes. The black pixels response to the low confident areas where only one directional gradient available, i.e., $\lambda_2/(\lambda_1 + \lambda_2) \approx 0$, which the large erroneous motion estimation vectors may occur.

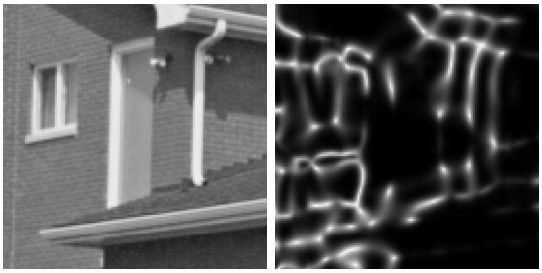
The lower performances of the gradient methods (GM and GOGM) can be attributed to its unequal treatment of spatial and temporal derivatives in the framework of the least-squares fitting (vertical offset), while those derivatives are treated equally in the tensor methods (GSTM and GOSTM) as they em-



(a) (b)

(c) (d)

Fig.6: *Estimated motion vectors of House image by using the (a) GM, (b) GOGM, (c) GSTM, (d) GOSTM.*



(a) (b)

Fig.7: *(a) A part of House image, (b) a visualization of confidence map of (a).*

ploy principal component analysis (PCA) or the orthogonal regression (orthogonal offset) [11]. Another reason for the difference is that the gradient methods estimate motions with only two frames, while the tensor-based methods do the task using three frames. Consequently, the GSTM and GOSTM work constantly better than the GM and GOGM.

4.2 Robustness to Time-Varying Intensities

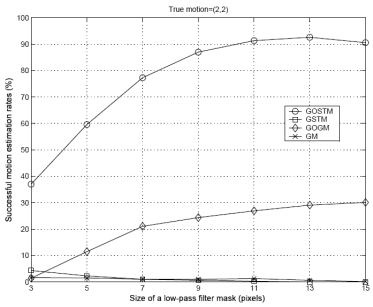
Next, we compare the robustness of the four techniques to temporal changes of image intensities. For this test, the intensities of the third frame of the tensor methods are uniformly reduced by 10%. On the other hand, the intensities of the second frame of the gradient methods are also reduced with the same ratio.

We obtain 28 by 28, i.e., 784 motion vectors in total per image sequence for the same five test images. Meanwhile, the size of Gaussian low-pass filters are varied and applied to the image sequence prior to motion estimation and also after image gradients are computed. The size of the local integration in Eqs. (14) and (15) is set to be 8 by 8 by 3 voxels for the GOSTM. The size of the local integration of Eqs. (12) and (13) is set 8 by 8 pixels.

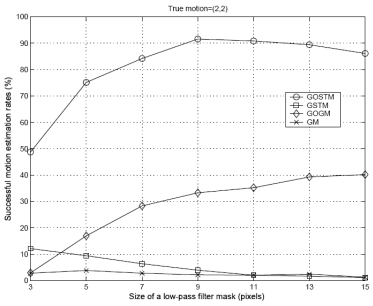
Fig. 8 shows the success rates of the four techniques with time-varying image intensities where the abscissa denotes the size of the low-pass filter mask used and the ordinate the successful motion estimation rates. It is apparent that the GM and GSTM are extremely sensitive to changes in image intensities, and do not work properly at all in such condition. Conversely, the GOGM and GOSTM perform motion estimation regardless of temporal changes of image intensities. This stark contrast clearly shows a significant advantage of the proposed methods, GOGM and GOSTM, over the conventional methods, GM and GSTM.

4.3 Motion Estimation on A Real Image Sequences

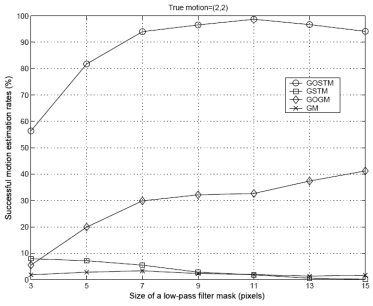
Finally, we compare the feasibility of the proposed methods on a real image sequences which contain local motions. Figs. 9 and 10 show real image se-



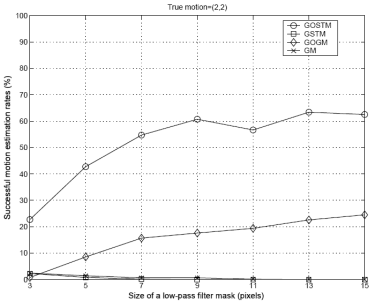
(a)



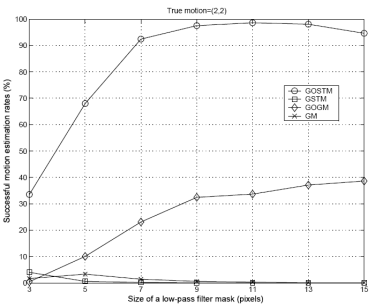
(b)



(c)



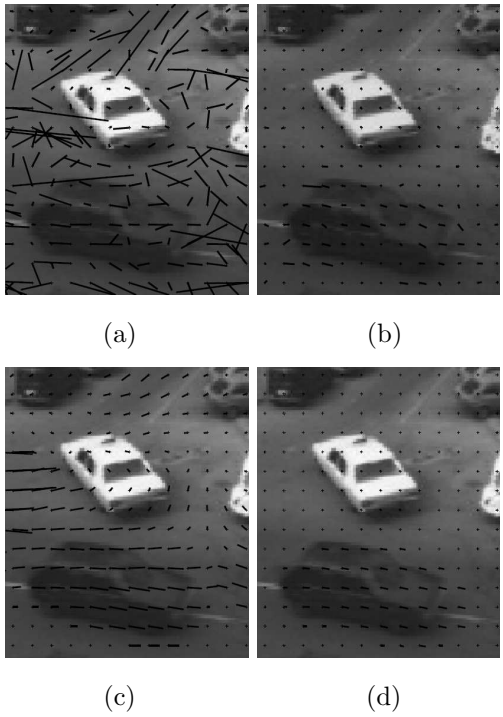
(d)



(e)

Fig.8: *Success rates for the (a) Lena, (b) Girl, (c) Pepper, (d) House, and (e) Mandril images when images intensities are changed 10% over time.*

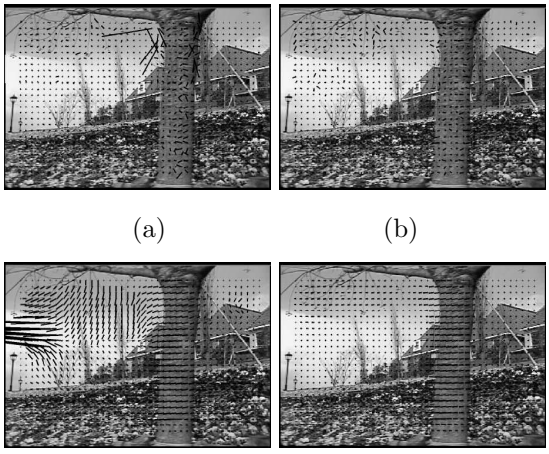
quences together with the motion vectors estimated by the four techniques. We again reduced the image intensities of one of the frames in the sequences uniformly by 10%. The size of the block is set at 8×8 pixels. Since there is no ground truth data for this sequence, the proposed methods GOGM and GOSTM are compared with the GM and GSTM based on visual inspection. As shown in Figs. 9 and 10, it is obvious that the conventional approaches the GM (Figs. 9(a), 10(a)) and GSTM (Figs. 9(c), 10(c)) cannot produce reliable motion estimates at all under the time-varying lighting condition. It should be noted that they produce numerous faulty motion estimates in background such as road and buildings where there is no motion. On the other hand, both GOGM (Figs. 9(b), 10(b)) and GOSTM (Figs. 9(d), 10(d)) work well under the same condition, and there are far less faulty responses in the background. Hence, it is confirmed that the proposed techniques outperform their conventional counterparts, especially under a time-varying illumination condition.



(a) (b)

(c) (d)

Fig.9: *Part of the motion estimation results on a real image sequence under a time-varying lighting condition by the (a) GM, (b) GOGM, (c) GSTM, and (d) GOSTM.*



(a) (b)

(c) (d)

Fig.10: *Part of the motion estimation results on a real image sequence, the flower garden, under a time-varying lighting condition by the (a) GM, (b) GOGM, (c) GSTM, and (d) GOSTM.*

5. CONCLUSIONS

This paper presents two novel motion estimation techniques using gradient orientation information. We refer to the two proposed methods as the gradient orientation based gradient method (GOGM) and the gradient orientation structure tensor method (GOSTM). They are based on the traditional gradient-based methods, the spatio-temporal gradient method (GM) and the gradient structure tensor method (GSTM). A comparative study among the four techniques shows that the proposed methods outperform the conventional approaches they are based on under a mildly noisy (SNR 40dB) and constant lighting condition. The improvement is achieved in the merging step of two motion estimates in which the weights are adapted depending on the diversity of the gradient directions in each local area (block). This step makes the proposed method robust to the aperture problem and huge erroneous motion estimates can be effectively suppressed. Another remarkable strength of the proposed methods is that they can perform motion estimation regardless of uniform changes of image intensities over time. Note that the conventional approaches do not work at all under such condition. This improvement is achieved by the use of gradient orientation that is known to be robust to changes in intensities. The paper also discusses the effect of the size of a preprocessing low-pass filter and a block where a motion vector is estimated. Simulation results clearly show that successful motion estimation rates increase as the size of the low-pass filter and the block increase. It should be noted, however, that the higher performance is achieved at the sacrifice of the resolution of the segmentation boundaries between two objects. From the simulation results, we address that a low-pass filters (typically Gaussian type) of size 11×11 or 13×13 pixels may be a good compromise between the accuracy and resolution of motion estimates. Meanwhile, the appropriate choice of the block size may be dependent on the application one tackles.

References

[1] B.K.P. Horn and B.G. Schunck, "Determining Optical Flow," *artificial intelligence*, 17, pp.185-203, 1981.

[2] A. Giachetti, "Matching techniques to compute image motion," *Image and Vision Computing*, vol. 18, pp. 247-260, 2000.

[3] P. Boonsieng and T. Kondo, "Comparative study of motion estimation techniques: the gradient method and structure tensor method," *Proceedings of the International Workshop on Advanced Image Technology*, Bangkok, Thailand, 2007.

[4] R.C. Gonzalez and R.E. Woods, *Digital Image Processing*, Prentice-Hall, Inc., 2002.

[5] M. Ye, R.M. Haralick, and L.G. Shapiro, "Estimating piecewise-smooth optical flow with global matching and graduated optimization," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 12, pp. 1625-1630, December 2003.

[6] C.-F. Westin and H. Knutsson, "Estimation of motion vector fields using tensor field filtering," *IEEE International Conference on Image Processing*, vol. 2, pp. 237-241, 1994.

[7] J. Zhang, J. Gao, and W. Liu, "Image sequence segmentation using 3-D structure tensor and curve evolution," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 11, No.5, pp. 629-641, 2001.

[8] H. Liu, R. Chellappa, and A. Rosenfeld, "Accurate dense optical flow estimation using adaptive structure tensors and a parametric model," *IEEE Trans. Image Processing*, vol. 12, no. 10, pp. 1170-1180, 2003.

[9] R. Jain, T. Kasturi, and B.G. Schunck, *Machine Vision*, McGraw-Hill, Inc., 1995.

[10] J.K. Kearney, W.B. Thompson, and D.L. Boley, "Optical flow estimation: an error analysis of gradient-based methods with local optimization," *IEEE Trans. Pattern Analysis Machine Intelligence*, vol. PAMI-9, no.2, pp.229-244, March 1987.

[11] T.R. Reed, *Digital Image Sequence Processing Compression, and Analysis*, CRC Press, United State of American, 2005.

[12] N. Nikokaidis and I. Pitas, *3-D Image Processing Algorithms*, Wiley/Interscience, 2001.

[13] R. Strzodka and C. Garbe, "Real-time motion estimation and visualization on graphics cards," *IEEE Visualization*, October 2004.

[14] R. Pless and J. Wright, "Analysis of persistent motion patterns using the 3D structure tensor," *Proceedings of the IEEE Workshop on Motion and Video Computing*, vol. 2, 2005.

[15] P.Y. Burgi, "Motion estimation based on the direction of intensity gradient," *Image and Vision Computing*, vol. 22, pp. 637-653, 2004.

[16] T. Kondo and H. Yan, "Automatic human face detection and recognition under non-uniform illumination," *Pattern Recognition*, vol. 32, pp. 1707-1718, 1999.

[17] J.L. Barron, D.J. Fleet, and S.S. Beauchemin, "Performance of Optical Flow Techniques." *International Journal of Computer Vision*, vol. 12, no. 1, pp.43-77, 1994.

Pramuk Boonsieng received the B.E. degree in Electrical Engineering in 2005 and M.S. degrees in Telecommunication in 2009, both from Sirindhorn International Institute of Technology, Thammasat University, Thailand. He was a research student at the University of Applied Sciences Ravensburg-Weingarten, Germany in 2007. He is studying toward his Ph.D. in Information, Computer, and Communication Technology at Sirindhorn International Institute of Technology, Thammasat University, Thailand. His research interests include optical network design, digital image processing and computer vision.

Toshiaki Kondo received his B.E. and M.E. degrees from the Tokyo Institute of Technology, Japan in 1986 and 1988, respectively. He was a research engineer for Canon Inc. from 1988 to 1999. He received his second M.E. degree from the University of Sydney, Australia in 1997. In 2004, he received his Ph.D. in image processing from National University of Singapore, Singapore. Since 2004, he has been with Sirindhorn International Institute of Technology, Thammasat University, Thailand where he is currently an Assistant Professor at School of Information, Computer, and Communication Technology. His research interests include image processing, computer vision, and pattern recognition. He is a member of The Institute of Electronics, Information and Communication Engineers (IEICE) and the IEEE Signal Processing Society.

Waree Kongprawechnon received her B.Eng with 1st class class honor from Chulalongkorn University in 1992. From 1992 to 1998, she got Japanese government to continue her graduate study in Japan. She received her M.Eng from Osaka University in 1995. She received her Ph.D from The University of Tokyo in 1998. Since 1998, she has joined Sirindhorn International Institute of Technology (SIIT), Thammasat University as a faculty member. Her research interests include H-infinity Control, Robust Control, Learning Control, Control Theory and its application.