A Wavelet-Based Approach for Color Image Registration

Yu-Te Wu,*1,2 Li-Fen Chen,2,6 Hui-Yun Chen,1,2 Po-Lei Lee,2 Tzu-Chen Yeh,2,4 and Jen-Chuen Hsieh2,4,5

1 Institute of Radiological Sciences, National Yang-Ming University, Taipei, Taiwan
2 Integrated Brain Research Laboratory, Department of Medical Research and Education, Taipei Veterans General Hospital, Taipei, Taiwan
3 Center for Neuroscience, National Yang-Ming University, Taipei, Taiwan
4 Faculty of Medicine, School of Medicine, National Yang-Ming University, Taipei, Taiwan
5 Institute of Neuroscience, School of Life Science, National Yang-Ming University, Taipei, Taiwan
6 Office of Research & Development, National Yang-Ming University, Taipei, Taiwan

We present a method to accurately estimate dense motion vectors between two successive color images in a sequence. This study integrates the red, green and blue channels of color images and extends the previous method30 in which the Cai–Wang wavelet representation was employed as a hierarchical motion model to match gray scaled images. The coarse-to-fine wavelet-based motion model is substituted into the objective function where the sum of squared differences (SSD) between two color images is minimized iteratively. Compared with the gray scaled images, our simulated experiments show that color images have better trackability and smaller condition numbers during the optimization process, leading to faster convergence rate and more accurate results. The estimated dense correspondences have been applied effectively to create realistic novel views using trilinear constraints.


Introduction

Image registration1–6 has been an important issue in computer vision and image processing for more than a decade, as many algorithms rely on accurate computation of correspondences through a sequence of images. For examples, in the applications of stereo matching, robot navigation, new view rendering, reconstruction of 3D scenes, and image segmentation,7–11,13,14 accurate estimation of the correspondence between two successive images is necessary for the automated and non-interactive uses of the implemented techniques. This article presents a coarse-to-fine, wavelet based method to accurately estimate dense motion vectors between two successive color images. The wavelet based method for matching grayscale images has been thoroughly tested with synthetic as well as real data set, and has been demonstrated its effectiveness in computing facial motion from video image sequences and disparity from stereo images.1,14 By taking the red (R), green (G) and blue (B) channels of color images into account, this work focuses on improving the accuracy of previous grayscale wavelet based method.

The motion vectors in the proposed algorithm are modeled by a linear combination of coarse-to-fine basis functions that are generated by dilating and translating two functions, a scaling function and a wavelet developed by Cai and Wang.15 A coarse-scale basis function has a large support while a fine-scale basis function has a small support. When the coarse-to-fine wavelet-based motion model is substituted into the sum of squared differences (SSD) to perform minimization process, the basis functions at different scales are used concurrently and served as large-to-small windows, which allow both the global and local information to be utilized simultaneously for image registration. In particular, the coefficients at the coarsest scale level are first estimated by using the basis functions with largest windows (in full resolution) to construct an approximation of motion vectors, and the coefficients from the coarsest to current fine scales are then estimated so that more details are added to refine the previous estimates. Accordingly, this method is not only suitable for recovering the motion vectors due to large motion, but also provides a solution to estimate the general motion, for example the nonrigid facial motion,2 in which the appropriate motion model and number of parameters are in general not known a priori. In addition, internal smoothness constraints are imposed via interpolation on each multi-resolution component to suppress noisy estimation. Note that the coarse-to-fine transform differs from the conventional wavelet transform17,18 that is carried out by two stages, that is from fine to coarse scales for decomposition and then from coarse to fine scales for reconstruction.

One way to improve the estimation of motion vectors further is the utilization of color images rather than gray-value ones in order to reduce the ambiguities. The
incorporation of color information is quite plausible since a blue pixel in one image should not correspond to red pixels in the other image. On the other hand, the intensity values in the grayscale representations of both images can be identical for the red and blue pixels. Thus, color information should allow better discrimination and higher probability of the Hessian matrices derived from the color image. For this reason, we investigated the color information in correspondence. Recent investigations using color information in stereo analysis and motion estimation have shown that, in comparison with gray value methods, an improvement of the results can be achieved. In this study, we applied the wavelet based method to both color images and their grayscale counterparts. In order to develop a coarse-to-fine wavelet representation in $H^2(I)$, Cai and Wang used the fourth-order B-spline scaling function $\phi(x)$ (Fig. 1(a))

$$
\phi(x) = \frac{1}{6} \sum_{j=0}^{4} \left( \frac{4}{j} \right) (-1)^j (x - j)^3_+ \in H^2(I)
$$

as the scaling function, where for any integer $n$

$$
x^n_+ = \begin{cases} x^n, & x \geq 0 \\ 0, & \text{otherwise} \end{cases}
$$

Using the scaling function, they constructed the wavelet function $\psi(x)$ (Fig. 1(b))

$$
\psi(x) = \frac{-3}{7} \phi(2x) + \frac{12}{7} \phi(2x - 1) + \frac{-3}{7} \phi(2x - 2).
$$

The supports of $\phi(x)$ and $\psi(x)$ are $[0, 4]$ and $[0, 3]$, respectively, i.e., the values of $\phi(x)$ (or $\psi(x)$) are zeros outside $[0, 4]$ (or $[0, 3]$). The dilation and translation of $\phi(x)$ and $\psi(x)$ are defined by

$$
\phi_j(x) = \phi(2^j x - k), \quad j \geq 0, \quad k = -2, \ldots, 2^j - 2
$$

$$
\psi_j(x) = \psi(2^j x - k), \quad j \geq 0, \quad k = -1, \ldots, 2^j - 2.
$$

Cai and Wang have shown that any continuous function in $H^2(I)$ can be approximated as closely as possible by the sum of linear combinations of $\phi(x - k)$ and $\psi(2^j x - k)$ for a sufficiently large resolution level $j$. It has been further demonstrated in Refs. 1 and 14 that it holds for any finite sampled function $d(x)$, which can be represented as follows:

$$
d(x) = d_{-1}(x) + d_0(x) + d_1(x) + \ldots + d_j(x)
$$
where

$$d_{-1}(x) = \sum_{k=-2}^{L-2} c_{-1,k} \phi_{0,k}(x)$$  \hspace{1cm} (6)$$

$$d_j(x) = \sum_{k=-2}^{2^j-2} c_{j,k} \psi_{j,k}(x) \text{ for } j \geq 0$$  \hspace{1cm} (7)$$

Note that the components $d_j(x)$ are computed from the coarsest resolution scale $j = -1$ to finer resolution scale in which the influence of coefficients becomes more local because the basis function has narrower support. $14$

**Coarse-to-Fine Wavelet based Motion Model**

Two-dimensional motion vectors $u(x,y)$ and $v(x,y)$ can be approximated by using two-dimensional basis functions. Tensor products are used to extend the basis functions from one to two dimensional contexts. Accordingly, the two-dimensional basis functions can be obtained as follows (Fig. 3):

$$\Phi_{0,k_1,k_2}(x,y) = \phi(x-k_1)\psi(y-k_2)$$  \hspace{1cm} (8)$$

$$\Psi^H_{j,k_1,k_2}(x,y) = \psi(2^jx-k_1)\phi(2^jy-k_2)$$  \hspace{1cm} (9)$$

$$\Psi^V_{j,k_1,k_2}(x,y) = \phi(2^jx-k_1)\psi(2^jy-k_2)$$  \hspace{1cm} (10)$$

$$\Psi^D_{j,k_1,k_2}(x,y) = \psi(2^jx-k_1)\psi(2^jy-k_2)$$  \hspace{1cm} (11)$$

where the subscripts $j$, $k_1$ and $k_2$ represent the resolution scale, horizontal and vertical translations, respectively, and the superscripts $H$, $V$ and $D$ represent the horizontal, vertical and diagonal directions, respectively. Similar to the one-dimensional case, any two-dimensional motion vector can be expressed in terms of the linear combinations of the coarsest scale spline function (8), the horizontal, vertical and diagonal wavelets ((9) (10) and (11)) in finer levels. $1$

$$u(x,y) = u_{-1}(x,y) + \sum_{j=0}^{L-2} \left[ u_j^H(x,y) + u_j^V(x,y) + u_j^D(x,y) \right]$$  \hspace{1cm} (12)$$

$$v(x,y) = v_{-1}(x,y) + \sum_{j=0}^{L-2} \left[ v_j^H(x,y) + v_j^V(x,y) + v_j^D(x,y) \right]$$  \hspace{1cm} (13)$$

where

$$u_{-1}(x,y)$$

$$= \sum_{k_1=-2}^{L_1-2} \sum_{k_2=-2}^{L_2-2} c_{-1,k_1,k_2} \Phi_{0,k_1,k_2}(x,y)$$

$$= \left[ c_{-1,-2,-2} \cdots c_{-1,L_1-2,L_2-2} \right]$$

$$[\Phi_{-2,-2}(x,y) \cdots \Phi_{L_1-2,L_2-2}(x,y)]^T$$

$$\text{def} \rightarrow \vec{c}_{-1}(x,y)$$

Figure 2. The translation and dilation of basis functions at different resolution scales.
Figure 3. The two dimensional basis functions. (a) $\psi(x)\phi(y)$; (b) $\phi(x)\psi(y)$; (c) $\psi(x)\phi(y)$; and (d) $\phi(x)\phi(y)$. A full color version of this figure can be found as Supplemental Material on the IS&T website (www.imaging.org) for a period of no less than two years from the date of publication.
Functions \( v_{x}, v_{y}, d_{x}, d_{y}, b_{x}, b_{y} \) and \( c_{x}, c_{y} \) have similar forms except that \( c_{-1, x, y}, d_{x}, d_{y}, b_{x}, b_{y} \) and \( c_{x}, c_{y} \) are replaced by \( d_{-1, x, y}, d_{x}, d_{y}, b_{x}, b_{y} \) and \( d_{x}, d_{y}, b_{x}, b_{y} \), respectively. We can concatenate all the coefficients defined in Eqs. (14) through (17) and rewrite Eq. (12) into a compact form

\[
\mathbf{u}(x, y) = \mathbf{v}(x, y)
\]

(18)

where

\[
\mathbf{v} = [c_{-1} \mathbf{v}_{0} \ldots c_{j} \mathbf{v}_{0} \ldots c_{J} \mathbf{v}_{0} \ldots c_{J} \mathbf{v}_{0} \ldots c_{J} \mathbf{v}_{0} \ldots c_{J}]
\]

(19)

Similarly, the vertical motion vectors can be expressed as

\[
\mathbf{v}(x, y) = \mathbf{d} \mathbf{w}(x, y)
\]

(21)

where

\[
\mathbf{d} = [d_{-1} \mathbf{d}_{0} \ldots d_{j} \mathbf{d}_{0} \ldots d_{J} \mathbf{d}_{0} \ldots d_{J} \mathbf{d}_{0} \ldots d_{J} \mathbf{d}_{0} \ldots d_{J}]
\]

(22)

and \( \mathbf{w} \) is given by Eq. (20).

**The Registration Algorithm**

A color image may have various formats, such as RGB (Red, Green, Blue), YIQ, CMY (Cyan, Magenta, Yellow), and HSI (Hue, Saturation, Intensity), and each one of them can be converted into another one. In this study, we use the RGB color images with 8 bits for each band. The corresponding gray scaled images are formed by weighted averaging 3 bands (0.299R + 0.587G + 0.114B) and thus have 8 bits for each pixel. Let us assume that two successive RGB color images are denoted by \( R_{x}, R_{y}, G_{x}, G_{y}, B_{x}, B_{y} \) and \( R_{x}, R_{y}, G_{x}, G_{y}, B_{x}, B_{y} \), respectively. Each band can be considered as a gray scaled image. The motion vectors between these two images are denoted by \( u(x, y), v(x, y) \), whose representations are given in Eqs. (12) and (13), and are estimated by minimizing the following SSD function

\[
E(u(x, y), v(x, y)) = \sum_{x, y} \left[ (R_{x}(x + u(x, y), y + v(x, y)) - R_{0}(x, y))^{2} + (G_{x}(x + u(x, y), y + v(x, y)) - G_{0}(x, y))^{2} + (B_{x}(x + u(x, y), y + v(x, y)) - B_{0}(x, y))^{2} \right].
\]

(23)

Under the assumption of intensity constancy, the intensity constraint for the motion vector \((u(x, y), v(x, y))\) can be written as

\[
I_{1}(x + u(x, y) + v) = I_{2}(x, y)
\]

(24)

where \( I \) represents \( R, G, \) or \( B \) bands of a color image. To simplify the notation, we drop the arguments \( x \) and \( y \) in \( u(x, y) \) and \( v(x, y) \) in subsequent derivations.

The SSD function (23) is in general not a quadratic function. We expand Eq. (24) into the first order Taylor series

\[
I_{1}(x + u(x, y) + v) = I_{1}(x, y) + uI_{x}(x, y) + vI_{y}(x, y)
\]

(25)

where

\[
I_{x}(x, y) = \frac{\partial I_{1}(x + u(x, y) + v)}{\partial x}, \quad I_{y}(x, y) = \frac{\partial I_{1}(x + u(x, y) + v)}{\partial y}
\]

and substitute this expansion into (23) to obtain a quadratic approximation

\[
E(u, v) = \sum_{x, y} \left[ (R_{1}(x, y) - R_{0}(x, y))^{2} + uR_{x}(x, y) + vR_{y}(x, y))^{2} + (G_{1}(x, y) - G_{0}(x, y))^{2} + uG_{x}(x, y) + vG_{y}(x, y))^{2} + (B_{1}(x, y) - B_{0}(x, y))^{2} + uB_{x}(x, y) + vB_{y}(x, y))^{2} \right].
\]

(26)

where

\[
(R_{x}, R_{y}) = \left( \frac{\partial R_{1}(x + u(x, y) + v)}{\partial x}, \frac{\partial R_{1}(x + u(x, y) + v)}{\partial y} \right),
\]

\[
(G_{x}, G_{y}) = \left( \frac{\partial G_{1}(x + u(x, y) + v)}{\partial x}, \frac{\partial G_{1}(x + u(x, y) + v)}{\partial y} \right),
\]

\[
(B_{x}, B_{y}) = \left( \frac{\partial B_{1}(x + u(x, y) + v)}{\partial x}, \frac{\partial B_{1}(x + u(x, y) + v)}{\partial y} \right).
\]

This quadratic SSD function can be further parameterized by substituting the wavelet models ((18) and (21)) into motion vectors \((u, v)\) and rewritten as follows

\[
E(c, d) = [c \mathbf{v} \mathbf{w}] A [c \mathbf{v} \mathbf{w}] \mathbf{T} - 2[c \mathbf{v} \mathbf{w}] b + \text{const}
\]

(27)

where

\[
A = \begin{bmatrix}
\sum_{x, y} (R_{x}^{2} + G_{x}^{2} + B_{x}^{2}) w w & \sum_{x, y} (R_{x} R_{y} + G_{x} G_{y} + B_{x} B_{y}) w w \\
\sum_{x, y} (R_{x} R_{y} + G_{x} G_{y} + B_{x} B_{y}) w w & \sum_{x, y} (R_{x}^{2} + G_{x}^{2} + B_{x}^{2}) w w
\end{bmatrix}
\]

(28)
Compute the wavelet Hessian matrix \( H \)\(^{29}\) as 
\[
\vec{b} = \left[ -\sum_{x,y} \left\{ \left[ R_x (R_1 - R_0) + G_x (G_1 - G_0) + B_x (B_1 - B_0) \right] w \right\} \\
-\sum_{x,y} \left\{ \left[ R_y (R_1 - R_0) + G_y (G_1 - G_0) + B_y (B_1 - B_0) \right] w \right\} \right]
\]

\[(29)\]

\[
\text{const} = \sum_{x,y} \left( (R_1 - R_0)^2 + (G_1 - G_0)^2 + (B_1 - B_0)^2 \right)
\]

\[(30)\]

The problem of motion estimation has been translated into a problem of estimating coefficients \( c \)'s and \( d \)'s in (19) and (22) such that the cost function (27) is minimized. The registration algorithm is designed to progressively recover the motion vectors from the coarsest resolution level such that the largest windows can be utilized to deal with large displacements. As the estimation proceeds to the finer resolution level, large-to-small windows are used and more details are computed to refine the previous estimate. The registration algorithm consists of the following steps:

1. Let \( \left\{ 2, \vec{d} \right\} \) only consist of coefficients at the coarsest resolution level \( (j=-1) \), all the coefficients are initialized to be zeros.
2. Compute the wavelet Hessian matrix \( A \) and gradient vector \( \vec{b} \).
3. Use the Levenberg–Marquardt method\(^{25}\) to minimize the SSD and compute the coefficients vector \( \left\{ 2, \vec{d} \right\} \).
4. Compute the pixelwise motion vectors \( (u,v) \) based on the estimated coefficients \( \left\{ 2, \vec{d} \right\} \) (19) and (22).
5. Use the pixelwise motion vectors \( (u,v) \) to warp color image \( (R_i(x + u, y + v), G_i(x + u, y + v), B_i(x + u, y + v)) \) toward \( (R_i(x, y), G_i(x,y), B_i(x,y)) \) based on the bilinear interpolation and compute the updated SSD.
6. If the SSD is smaller than a given threshold, the optimization at the current resolution scale is terminated and continued to the next finer resolution scale. The new coefficients vector consists of the coefficients obtained from the coarsest to the previous resolution scale and coefficients with zero initials at the new resolution scale. If the SSD is larger than a given threshold, step 2 to step 5 are repeated.
7. Repeat step 2 to step 6 until the pre-defined resolution scale is reached.

The structure of the wavelet Hessian matrix reveals the use of both global and local information and how they interact with each other. Each entry in the wavelet Hessian matrix is produced by multiplications of two basis functions and image gradient components. The entry is non-zero when the associated two basis functions are overlapped and image gradients are non-zero. As the resolution level moves up, the basis functions have narrower supports and there are fewer overlapping basis functions in coarser and current levels. The block diagonal non-zero entries are produced by two overlapped basis functions at the same scale level and the other non-zero blocks are produced by the overlaps of different basis functions from the coarsest to the current scales. As a result, in the finer scales not only the smaller but also the larger patches (windows) are concurrently used to optimize the finer- and coarser-scale coefficients.

**Trackability and Convergence**

The image Hessian matrix, \( H \), is an \( 2 \times 2 \) matrix formed by image gradients as follows

\[
H = \begin{bmatrix}
\sum_{(x,y)\in W} \left( \frac{\partial I}{\partial x} \right)^2 & \sum_{(x,y)\in W} \left( \frac{\partial I}{\partial x} \right) \left( \frac{\partial I}{\partial y} \right) \\
\sum_{(x,y)\in W} \left( \frac{\partial I}{\partial y} \right) \left( \frac{\partial I}{\partial x} \right) & \sum_{(x,y)\in W} \left( \frac{\partial I}{\partial y} \right)^2
\end{bmatrix}
\]

\[(31)\]

where \( W \) denotes the region of interest and \( I \) is used to represent \( R \) band, \( G \) band, \( B \) band, or gray-scale images. The eigenvalues of image Hessian matrices (31) and wavelet Hessian matrix (28) for each color image pair and its gray scaled counterpart are computed for the analysis of texture and convergence rate, respectively. For the problem of point feature tracking, the two eigenvalues of matrix \( H \) have been used as a measure of “trackability” for the selection of good points where \( W \) is a small image window centered on each point. Specifically, to determine whether a region of the image constitutes a trackable feature point, two eigenvalues of the image Hessian matrix are examined. Relatively textureless regions, which are difficult to track, usually have very low image derivatives and their image Hessian matrices have two small eigenvalues. Regions containing high spatial derivatives in one direction such as lines or edge features, tends to have one large and one small eigenvalues. When both eigenvalues are large, the regions have high spatial gradient in two orthogonal directions, and therefore can be tracked more reliably. On the other hand, following the discussions in Refs. 28, 29, and 30, the inverses of the two eigenvalues of \( H \) represent the degree of uncertainty of image regions. It has been addressed\(^{29}\) that under conditions of small Gaussian noise, the inverse eigenvalues are proportional to the variance in the motion estimates. As the eigenvalues are larger, the images are more trackable. In our study, since we estimate the dense correspondences rather than sparse feature points, the whole image is considered as the region of interest, \( W \), so that two eigenvalues represent the overall “trackability”.

The convergence of wavelet Hessian matrix can be analyzed from a mathematical point of view. When \( Q \) is a positive definite symmetric \( n \times n \) matrix and the quadratic function

\[
f(\vec{x}) = \vec{x}^T Q \vec{x} - \vec{2b}
\]

is minimized by using the steepest descent method, the condition number \( \lambda_\text{max}/\lambda_\text{min} \) of the largest to the smallest eigenvalues) of \( Q \) governs the convergence rate.\(^{31}\) It has been proven that the smaller is the ratio, the faster is the convergence of the steepest descent.\(^{31}\) Intuitively, if \( Q \) has one or more small eigenvalues, it is
nearly singular and the computation of its inverse, as required in the SSD optimization process for solving a linear system, becomes an ill conditioned problem. When both eigenvalues are large, the matrix is invertible so the linear equation is well-conditioned. Since we apply the Levenberg–Marquardt method, which combines the steepest descent and the inverse-Hessian methods, to minimize the quadratic SSD (Eq. (27)), the convergence is dominated by the steepest descent. Therefore, the condition numbers obtained from color and gray-scale wavelet Hessian matrices are used as indexes in the comparison of convergence rates.

Simulated Experiments

In this section, four simulated experiments were exemplified to demonstrate the advantage of color image pairs. Each color image has three bands (RGB) with 8 bits for each band and its corresponding gray scaled image is 8 bits. For all examples, \( L \) was set to 4 and the motion vectors were computed up to three resolution levels, i.e., \( j = 0, 1, 2 \). Two images in each color (or corresponding gray scaled) image pair differ from hypothetical displacements. The left image in the first color image pair (Fig. 4(a)) was produced by translating the right image (Fig. 4(b)) 5.5 pixels leftward and 4.0 upward. The second, third and fourth color image pairs are shown in Figs. 5, 6 and 7, respectively. The left images of the second, third and fourth examples (Figs. 5(a), 6(a) and 7(a)) were generated by warping the right image (Figs. 5(b), 6(b) and 7(b)) using the affine motion model defined by

\[
\begin{align*}
  u(x, y) &= a_1 x + a_2 y + a_3 - x \\
  v(x, y) &= a_4 x + a_5 y + a_6 - y
\end{align*}
\]  

(33)

where the parameters \((a_1, a_2, a_3, a_4, a_5, a_6)\) are set to be \((\cos(4^\circ), -\sin(4^\circ), 0.5, \sin(4^\circ), \cos(4^\circ), 1)\) \((\cos(3^\circ), -\sin(3^\circ), 0, \sin(3^\circ), \cos(3^\circ), 0)\) and \((\cos(2^\circ), -\sin(2^\circ), 2, \sin(2^\circ), \cos(2^\circ), 3)\), respectively. The size of the image in the third example is \(193 \times 193\) and that for other examples is \(257 \times 257\). The averaged hypothetical displacements for the second, third and fourth examples are 8.7, 6.4 and 7.8 pixels, respectively.

We calculated the \(2 \times 2\) matrices, \(H\) (Eq. (31)), by using the whole image as the region of interest, \(W\), and computed two eigenvalues for the left image of each color image pair and its gray scaled counterpart. Table I shows that color images do provide better trackability than their gray scaled counterpart since one of color bands always has larger eigenvalues \(\lambda_{\text{max}}\) and \(\lambda_{\text{min}}\). For instance, the red band in the forth example has larger \(\lambda_{\text{max}} = 8.73 \times 10^6\) and \(\lambda_{\text{min}} = 8.14 \times 10^6\) than \(\lambda_{\text{max}} = 8.6 \times 10^6\) and \(\lambda_{\text{min}} = 8.1 \times 10^6\) in the gray scaled one.

On the other hand, we also computed the logarithmic ratios of the condition number

\[
\log \left( \frac{\lambda_{\text{max}}}{\lambda_{\text{min}}} \right)
\]

of Hessian matrix \(A\) (Eq. (28)) at each iteration. Figures 4(e)–7(e) show that color images have smaller condition numbers (solid curves with cross marks) than gray scaled ones (dotted curves with circles), and produce faster convergent rates. Note that, when the resolution level increases from \(j = -1\) to \(j = 0\) and from \(j = 0\) and \(j = 1\), the ratios jump from one level to another level due to the increasing dimension of Hessian matrix \(A\) that enlarges the range of eigenvalues.

To evaluate the accuracy of the estimated motion vectors, we computed the angular errors, denoted by \(\theta_e\), and the magnitude errors, denoted by \(m_e\). The formula for \(\theta_e\) and \(m_e\) are defined by

\[
\theta_e = \arccos \left( \frac{\tilde{u}_t + \tilde{v}_t + 1}{\sqrt{\tilde{u}_t^2 + \tilde{v}_t^2 + 1}} \right)
\]  

(35)

\[
m_e = \sqrt{\tilde{u}_t^2 + \tilde{v}_t^2} - \sqrt{u_t^2 + v_t^2}
\]  

(36)

where \((\tilde{u}, \tilde{v})\) and \((u_t, v_t)\) represent the estimated and true motion vectors respectively. Table II summarizes the averaged angular error, \(\bar{\theta}_e\) (degree), and the averaged magnitude error \(\bar{m}_e\) (pixel) for four simulated experiments. In addition, the logarithms of averaged angular error and averaged magnitude error at each iteration are plotted in Figs. 4(f)–(g), 5(f)–(g), 6(f)–(g) and 7(f)–(g) for each example, respectively. The final results are summarized in Table II. Overall, the results obtained from color images have much smaller errors compared with that from gray scaled images. The correct and estimated motion vectors are plotted in Figs. 4(c)–(d), 5(c)–(d), 6(c)–(d) and 7(c)–(d), respectively.

Application to New View Synthesis

New view synthesis is a rendering technique for creating new views from limited images by placing a virtual camera at arbitrary locations. The principle of new view synthesis is based on the trilinear constraint to reproject the corresponding image points from reference images into a new view. The coefficients of trilinear equations are expressed in terms of image correspondences and the camera intrinsic and extrinsic parameters. Accordingly, the computation of accurate image correspondences and the camera calibration (or self-calibration) are indispensable to create such a novel view.

In this section, we present the result of new view synthesis using two color views of an object taken from known positions. The procedure in creating the new view is as follows. First, we utilized a planar calibration plate and applied the Faugeras’s approach to estimate the intrinsic and extrinsic parameters of camera. Second, the dense correspondences between two reference color images were estimated using the wavelet based method. Third, we specified the location of the virtual camera and used the dense correspondences as well as the intrinsic and extrinsic camera parameters to synthesize a new view based on the algebraically trilinear relation between two reference views and the new view.

We used a toy giraffe as the object in our experiments. Two references images are shown in Fig. 8 where the two locations of the object are different by 5 degrees around the vertical axis. Only color image pairs were processed because they have better trackability (see Table I). The resultant motion vectors are superimposed on the top of one of the reference images and are shown in the upper left plot of Fig. 8. The average of motion vectors is up to 13.5 pixels. The warped image and residual intensity differences are shown in the lower left panel and lower right panel, respectively. According to the resultant residual differences, most of the matchings are accurate except on the boundary of the object where there are no correspondences due to occlusions. The logarithmic scale of the condition numbers at each iteration
**Figure 4.** Simulated experiment 1: (a) Image $I_0$; (b) Image $I_1$; (c) True flow; (d) Estimated flow; (e) Logarithm of the condition numbers at each iteration of optimization process using the color image (solid curve with cross marks) and the corresponding gray scaled image (dotted curve with circles); (f) Logarithm of the averaged angular error at each iteration using the color image (solid curve with cross marks) and corresponding gray scaled image (dotted curve with circles); (g) Logarithm of the averaged magnitude error at each iteration using the color image (solid curve with cross marks) and corresponding gray scaled image (dotted curve with circles). A full color version of this figure can be found as Supplemental Material on the IS&T website (www.imaging.org) for a period of no less than two years from the date of publication.
Figure 5. Simulated experiment 2: (a) Image $I_0$; (b) Image $I_1$; (c) True flow; (d) Estimated flow; (e) Logarithm of the condition numbers at each iteration of optimization process using the color image (solid curve with cross marks) and the corresponding gray scaled image (dotted curve with circles); (f) Logarithm of the averaged angular error at each iteration using the color image (solid curve with cross marks) and corresponding gray scaled image (dotted curve with circles); (g) Logarithm of the averaged magnitude error at each iteration using the color image (solid curve with cross marks) and corresponding gray scaled image (dotted curve with circles). A full color version of this figure can be found as Supplemental Material on the IS&T website (www.imaging.org) for a period of no less than two years from the date of publication.
Figure 6. Simulated experiment 3: (a) Image $I_0$; (b) Image $I_1$; (c) True flow; (d) Estimated flow; (e) Logarithm of the condition numbers at each iteration of optimization process using the color image (solid curve with cross marks) and the corresponding gray scaled image (dotted curve with circles); (f) Logarithm of the averaged angular error at each iteration using the color image (solid curve with cross marks) and corresponding gray scaled image (dotted curve with circles); (g) Logarithm of the averaged magnitude error at each iteration using the color image (solid curve with cross marks) and corresponding gray scaled image (dotted curve with circles). A full color version of this figure can be found as Supplemental Material on the IS&T website (www.imaging.org) for a period of no less than two years from the date of publication.
Figure 7. Simulated experiment 4: (a) Image $I_0$; (b) Image $I_1$; (c) True flow; (d) Estimated flow; (e) Logarithm of the condition numbers at each iteration of optimization process using the color image (solid curve with cross marks) and the corresponding gray scaled image (dotted curve with circles); (f) Logarithm of the averaged angular error at each iteration using the color image (solid curve with cross marks) and corresponding gray scaled image (dotted curve with circles); (g) Logarithm of the averaged magnitude error at each iteration using the color image (solid curve with cross marks) and corresponding gray scaled image (dotted curve with circles). A full color version of this figure can be found as Supplemental Material on the IS&T website (www.imaging.org) for a period of no less than two years from the date of publication.
TABLE I. The Largest and Smallest Eigenvalues of Matrix $H$ for Gray Scaled and Color Images in Simulated (Ex.1–Ex.4) and Real Experiments (Giraffe and Colonoscopy)

<table>
<thead>
<tr>
<th>Examples</th>
<th>$\lambda_{\text{max}}$ ($\times 10^6$)</th>
<th>$\lambda_{\text{min}}$ ($\times 10^6$)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>G</td>
</tr>
<tr>
<td>Ex. 1: Translational motion</td>
<td>20.4</td>
<td>19.5</td>
</tr>
<tr>
<td>Ex. 2: Affine motion 1</td>
<td>6.96</td>
<td>6.86</td>
</tr>
<tr>
<td>Ex. 3: Affine motion 2</td>
<td>7.13</td>
<td>8.80</td>
</tr>
<tr>
<td>Ex. 4: Affine motion 3</td>
<td>8.60</td>
<td>8.73</td>
</tr>
<tr>
<td>Giraffe</td>
<td>1.72</td>
<td>1.42</td>
</tr>
<tr>
<td>Colonoscopy</td>
<td>6.03</td>
<td>6.09</td>
</tr>
</tbody>
</table>

TABLE II. The Averaged Angular and Magnitude Errors $(\bar{\theta}_e, \bar{m}_e)$ of Simulated Experiments

<table>
<thead>
<tr>
<th>Example</th>
<th>Gray scaled</th>
<th>Color</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ex 1: Translational motion</td>
<td>(0.0145, 0.1287)</td>
<td>(0.0003, 0.0053)</td>
</tr>
<tr>
<td>Ex 2: Affine motion 1</td>
<td>(0.0136, 0.3058)</td>
<td>(0.0034, 0.1494)</td>
</tr>
<tr>
<td>Ex 3: Affine motion 2</td>
<td>(0.0208, 0.1212)</td>
<td>(0.0005, 0.0053)</td>
</tr>
<tr>
<td>Ex 4: Affine motion 3</td>
<td>(0.0239, 0.0813)</td>
<td>(0.0154, 0.0557)</td>
</tr>
</tbody>
</table>

Figure 8. Application to new view synthesis. The upper left panel shows the estimated motion vectors superimposed on top of one reference image and the upper right panel shows the other reference image. The lower left panel shows the warped image using the estimated motion vectors and the reference image in the upper right panel. The lower right panel plots the intensity differences between the warped image and the other reference image in the upper left panel. A full color version of this figure can be found as Supplemental Material on the IS&T website (www.imaging.org) for a period of no less than two years from the date of publication.
Figure 9. The Logarithm of the condition numbers at each iteration of optimization process using the color image (solid curve with cross marks) and the corresponding gray scaled image (dotted curve with circles).

Figure 10. The results of new view synthesis using dense correspondences and trilinear constraint. The two reference images (marked by bold frames) are different by 5 rotational degrees. A full color version of this figure can be found as Supplemental Material on the IS&T website (www.imaging.org) for a period of no less than two years from the date of publication.
Figure 11. Colonoscopy images. The upper left panel shows the estimated motion vectors superimposed on top of one colonoscopy image and the upper right panel shows the other colonoscopy image. The lower left panel shows the difference between two colonoscopy images. The lower right panel plots the differences between the warped image and the other image in the upper left panel. A full color version of this figure can be found as Supplemental Material on the IS&T website (www.imaging.org) for a period of no less than two years from the date of publication.

Figure 12. The logarithm of the condition numbers at each iteration of optimization process using the color image (solid curve with cross marks) and the corresponding gray-scaled image (dotted curve with circles).
of optimization process using the color image (solid curve with cross marks) and the corresponding gray scaled image (dotted curve with circles) is depicted in Fig. 9. It is clear that the color image has much smaller condition number compared with its gray scaled one. Figure 10 shows the results of new view synthesis under rotation around the vertical axis. The two reference images are marked by the bold frame and the rest of the images are synthesized by rotating the vertical-axis of virtual camera coordinate every 5 degrees and a translation. The viewing range of the upper left and lower right synthesized images are 30 degrees away from the viewing cones of first and the second reference views, respectively. The averaged synthesis error for each view was less than one pixel.

The last experiment shows two image frames (the first row in Fig. 11) of a colonoscopy video clip in which the image motion was caused by the movement of colonoscope and potential colon deformation. The left image in the second row is the original differences between two image frames (we show them only in gray scale), and the right one is the residual difference after image registration. Clearly, these two images have been registered accurately to reduce most of the differences. Compared with the gray scaled image, the color one has superior trackability (Table I) and smaller condition number at each iteration (Fig. 12). When the camera parameters of the colonoscope are available, we can apply the techniques of the new view synthesis, or structure from motion technique to the estimated dense correspondences and build up an image-guided virtual-colonoscopy system.

Conclusions
We have presented a wavelet based method for color image registration. The proposed approach utilizes a hierarchical wavelet representation as a motion model with the following merits: (a) the motion vectors are reconstructed from coarse to fine for estimating large displacement (b) the basis functions are used as windows so that large-to-small full resolution regions are selected for image comparison and (c) smoothness constraint is imposed by interpolation. The red, green and blue bands of color images have been taken into account to minimize the SSD during the minimization process. To compare the overall trackability and convergence of the registration algorithm between color images and their gray scaled counterparts, we calculated the largest and smallest eigenvalues of the image Hessian matrices and condition numbers of the wavelet Hessian matrices. The superior results obtained from color images than gray scaled ones are due to larger eigenvalues in one of the color band and smaller condition numbers. We have applied the accurate dense correspondences to synthesize realistic, color new view based on trilinear constraints. The wavelet based method can be further employed in other applications of computer vision where accurate image correspondences are demanded.

Acknowledgement. This work has been supported in part by the National Science Council under Grants NSC9-2218-E-010-001, by the Ministry of Education, Taiwan, under Grants MOE89-B-FA22-1-401.

References