

# Phase-Based Feature Matching under Illumination Variances

Masaaki Nishino, Atsuto Maki, and Takashi Matsuyama

Graduate School of Informatics, Kyoto University  
nishino@mbbox.kudpc.kyoto-u.ac.jp

**Abstract.** The problem of matching feature points in multiple images is difficult to solve when their appearance changes due to illumination variance, either by lighting or object motion. In this paper we tackle this ill-posed problem by using the difference of local phase which is known to be stable to a certain extent even under illumination variances. In order to realize a precise matching, we basically compute the local phase by convolutions with Gabor filters which we design in multi scales. We then evaluate the stability of local phase against lighting changes. Through experiments using both CG and real images that are with illumination variance, we show the relevancy of our theoretical investigations.

**Key words:** computer vision, feature matching, local phase, Gabor filters, illumination.

## 1 Introduction

The problem of matching feature points in multiple images is important as a prerequisite for structure-from-motion or 3D reconstruction from multi-view images. Feature points are usually defined where high image gradients are observed in various directions so that they can be differentiated from their neighboring image points, and typically detected by operators [1–3] including Harris corner detectors. The task of feature matching is then to determine the set of corresponding points between input images by comparing the local intensity distributions. Given an image sequence,  $I^{(k)}$  ( $k = 1, 2, \dots$ ), that is due to relatively small motion from frame to frame, template matching is often employed by referring the intensities in small region around a feature point in  $k$ -th image,  $I^{(k)}$ , to find the corresponding point in the subsequent image,  $I^{(k+1)}$ .

One of the difficulties in finding the correspondence is to deal with illumination variance which tends to occur on the surface of a target object, in particular when it is in motion relative to the light sources. Namely, when the appearance of an object changes, in principle it is nearly impossible to accurately match a feature point in one image to another only by the direct comparison of intensity distributions. Some matches that are obviously wrong can be excluded as outliers by RANSAC [10], by investigating the consistency of matches as a group. However, the remaining matches would still suffer from drifts of feature points induced by the changes in their appearance.

The principal aim of this paper is to introduce a phase-based method for feature matching in order to cope with the issue of the drift under illumination variance. This is motivated for example by previous work of Fleet and Jepson [4] which shows that phase is amplitude invariant and robust with respect to smooth shading and lighting variation. Because of the characteristics, phase has been successfully applied to estimate optical flow [5], or stereo disparity in sub-pixel accuracy [6, 7]. Also, Carneiro and Jepson [8] have recently proposed a phase-based local feature. They have empirically shown its improved performance over differential invariants when dealing with common illumination changes.

In the remainder of this paper, we propose to carry out feature matching by using local phase of input image which we compute by convolutions with Gabor filters. We design the filters in various forms both in terms of scales and directions, and realize a precise matching by effectively combine the outputs. We also investigate the relevancy of using local phase for feature matching theoretically on the basis of a shading model of image intensities. In the experiments we evaluate our propositions in comparison with conventional template matching, using both CG and real images with illumination variance.

## 2 Preliminary

**Local Phase.** This section provides an overview of the phase-based matching method using 1D signal. The discussion is extended to the case of 2D images in the next section.

For matching feature points we use local phase which we compute by convolutions of Gabor filters with the input signal. Gabor filter is a complex-valued function and has a form at point  $x$ ,

$$g(x; \sigma, \omega_0) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-x^2}{2\sigma^2}} (e^{i\omega_0 x} - e^{\frac{-(\sigma\omega_0)^2}{2}}) , \quad (1)$$

where  $\sigma$  is a parameter which determines the width of the filter and  $\omega_0$  the central frequency of the power spectrum, respectively.<sup>1</sup> The term  $e^{-(\sigma\omega_0)^2/2}$  is added to remove the DC component of the filter. Let  $c(x_0, \omega_0)$  be a convolution of real-function,  $f(x)$ , and Gabor filter,  $g(x; \omega_0)$ , at point  $x_0$ , i.e.  $c(x_0, \omega_0) = (f * g)(x_0, \omega_0)$ . Since  $c(x_0, \omega_0)$  is a complex function, we can formulate it as

$$c(x_0, \omega_0) = \rho(x_0, \omega_0) e^{i\phi(x_0, \omega_0)} , \quad (2)$$

by using two real functions,  $\rho(x_0, \omega_0)$  and  $\phi(x_0, \omega_0)$ , which represent local amplitude and phase, respectively.

**The Principle of Phase-Based Matching.** Matching feature points in two images,  $I^{(1)}$  and  $I^{(2)}$ , by using local phase is basically to find such points that

---

<sup>1</sup> In below,  $\sigma$  is varied together with  $\omega_0$  so as to maintain the relation,  $\sigma\omega_0 = \pi$ .

have equivalent distributions of local phase. The spatial shift of the signal,  $d$ , is related to the phase difference,  $\Delta\phi(\omega_0)$ , by

$$d = \frac{\Delta\phi(\omega_0)}{\omega_0} . \quad (3)$$

Since  $\Delta\phi$  is limited to the range  $-\pi \leq \Delta\phi(\omega_0) \leq \pi$ ,  $d$  is also limited to

$$\frac{-\pi}{\omega_0} \leq d \leq \frac{\pi}{\omega_0} . \quad (4)$$

Our strategy is to first compute the phase at a feature point  $z^{(1)}$  in image  $I^{(1)}$  and then at a point  $z^{(2)'}$  in  $I^{(2)}$ , which is a candidate to match  $z^{(1)}$ . Taking the phase difference,  $\Delta\phi(\omega_0)$ , we can directly derive  $d$  by (3), which is the residual distance between  $z^{(2)'}$  and the point  $z^{(2)}$  that precisely corresponds to  $z^{(1)}$ . Hence, we can match feature points by the local phase.

Although the discussion above assumes that local phase does not change between two images, we revisit the relevancy of the assumption in Sect. 4.

### 3 Phase-Based Feature Matching in 2D Images

#### 3.1 Convolutions with 2D Gabor Filters

To detect local phase in 2D images, we use 2D Gabor filters,  $g(x, y)$ , which are

$$g(x, y; \omega_0, \theta_0) = \frac{1}{2\pi\sigma^2} e^{\frac{-1}{2\sigma^2}(x^2+y^2)} (e^{i\omega_0 x} - e^{\frac{-(\sigma\omega_0)^2}{2}}) , \quad (5)$$

where  $\dot{x}, \dot{y}$  are defined as

$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} \cos\theta_0 & \sin\theta_0 \\ -\sin\theta_0 & \cos\theta_0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} , \quad (6)$$

and  $\theta_0$  and  $\omega_0$  denote the direction and the central frequency of the filter, respectively.

Here, we give a theoretical interpretation to the convolution of a 2D function and a 2D Gabor filter. Equation (5) can be expressed as a product of two components,  $g_{\dot{x}}(\dot{x}; \omega_0, \theta_0)$  and  $g_{\dot{y}}(\dot{y}; \omega_0, \theta_0)$ , which are defined to be

$$g_{\dot{x}}(\dot{x}; \omega_0, \theta_0) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-\dot{x}^2}{2\sigma^2}} (e^{i\omega_0 \dot{x}} - e^{\frac{-(\omega_0\sigma)^2}{2}}) , \quad \text{and} \quad (7)$$

$$g_{\dot{y}}(\dot{y}; \omega_0, \theta_0) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-\dot{y}^2}{2\sigma^2}} , \quad (8)$$

respectively. Using them, we can represent  $(f * g)(x_0, y_0)$ , the convolution of a 2D signal  $f(x, y)$  and a 2D Gabor filter  $g(x, y)$  at point  $(x_0, y_0)$ , as

$$\begin{aligned} (f * g)(x_0, y_0) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) g(x_0 - x, y_0 - y) dx dy \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \dot{f}(\dot{x}, \dot{y}) g_{\dot{x}}(\dot{x}_0 - \dot{x}) g_{\dot{y}}(\dot{y}_0 - \dot{y}) d\dot{x} d\dot{y} \\ &= \int_{-\infty}^{\infty} g_{\dot{x}}(\dot{x}_0 - \dot{x}) \left\{ \int_{-\infty}^{\infty} \dot{f}(\dot{x}, \dot{y}) g_{\dot{y}}(\dot{y}_0 - \dot{y}) d\dot{y} \right\} d\dot{x} , \quad (9) \end{aligned}$$

where  $(\dot{x}_0, \dot{y}_0)$  corresponds to point  $(x_0, y_0)$  on  $\dot{x}$ - $\dot{y}$  coordinate system such that  $\dot{x}_0 = x_0 \cos \theta_0 + y_0 \sin \theta_0$ , and  $\dot{y}_0 = -x_0 \sin \theta_0 + y_0 \cos \theta_0$ , respectively.  $\dot{f}(\dot{x}, \dot{y})$  also corresponds to the signal  $f(x, y)$  in this coordinate system, i.e.  $\dot{f}(\dot{x}, \dot{y}) = f(x, y)$ . Now, defining the following function of  $\dot{x}$ ,

$$\dot{f}_{\theta_0}(\dot{x}) = \int_{-\infty}^{\infty} \dot{f}(\dot{x}, \dot{y}) g_{\dot{y}}(\dot{y}_0 - \dot{y}) d\dot{y} , \quad (10)$$

we can interpret (9) as the convolution of the 1D function  $\dot{f}_{\theta_0}(\dot{x})$  and a 1D Gabor filter,  $g_{\dot{x}}$ . We call  $\dot{f}_{\theta_0}(\dot{x})$  the *directional function* for direction  $\theta_0$ .  $\dot{f}_{\theta_0}(\dot{x})$  turns out to be a 1D function obtained by convolving a 2D function  $\dot{f}(\dot{x}, \dot{y})$  with the Gauss function,  $g_{\dot{y}}$ , at each point on line  $\dot{y} = \dot{y}_0$  for the direction of  $\dot{y}$ .

### 3.2 Issues in 2D Phase-Based Matching

In 2D images, the relation between the difference of local phase and the distance of two points is formulated as

$$\mathbf{d} \cdot \boldsymbol{\omega}_0 = \Delta\phi(\boldsymbol{\omega}_0) , \quad (11)$$

where  $\mathbf{d} = (d_x, d_y)$  and  $\boldsymbol{\omega}_0 = (\omega_0 \cos \theta_0, \omega_0 \sin \theta_0)$  are both 2-vector and  $\mathbf{d}$  represents the distance, while  $\boldsymbol{\omega}_0$  the central frequency of the Gabor filters. However, different from the case in 1D signal, we cannot determine  $\mathbf{d}$  directly from  $\boldsymbol{\omega}_0$  and  $\Delta\phi(\boldsymbol{\omega}_0)$  since  $\Delta\phi(\boldsymbol{\omega}_0)$  is a scalar. Therefore we first assume that  $\mathbf{d}'(\theta_0)$  is the projection of  $\mathbf{d}$  to the direction  $\theta_0$ , and substitute  $\mathbf{d}'$  for  $\mathbf{d}$  in (11). With this, we can determine  $\mathbf{d}'(\theta_0)$  directly, and then calculate  $\mathbf{d}$  using two estimates of  $\mathbf{d}'(\theta_0)$  for different directions. This is valid when the 2D signal has simple form, but not always in real images. Moreover, the local phase may become unstable when the real component of the Fourier transform of the signal is small.

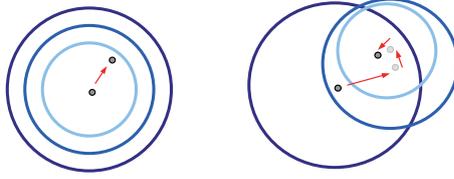
### 3.3 Integration of Filters by Using Evaluation Function

For the reasons stated above, we employ multiple Gabor filters to compute the distance,  $\mathbf{d}$ , and integrate the outputs of these filters. We then apply an evaluation function,  $J_{\mathbf{x}, \mathbf{x}'}(\mathbf{d})$ , of the distance between two points  $\mathbf{x}$  and  $\mathbf{x}'$  that Wiskott et al. proposed [9] for solving a graph matching problem. When we use  $N$  different Gabor filters that have different combination of parameters  $\omega_0$  and  $\theta_0$ , we have

$$J_{\mathbf{x}, \mathbf{x}'}(\mathbf{d}) = \sum_{j=1}^N \rho_j(\mathbf{x}) \rho_j(\mathbf{x}') \{ \Delta\phi_j(\mathbf{x}, \mathbf{x}') - \mathbf{d} \cdot \boldsymbol{\omega}_j \}^2 , \quad (12)$$

where  $\Delta\phi_j(\mathbf{x}, \mathbf{x}') = \phi_j(\mathbf{x}) - \phi_j(\mathbf{x}')$ , and  $\boldsymbol{\omega}_j = (\omega_{jx}, \omega_{jy}) = (\omega_j \cos \theta_j, \omega_j \sin \theta_j)$ . Since the term  $\Delta\phi_j(\mathbf{x}, \mathbf{x}') - \mathbf{d} \cdot \boldsymbol{\omega}_j$  is a transformation of (11) and should be zero for proper  $\mathbf{d}$  in an ideal case, we can obtain  $\mathbf{d}$  as minimizing  $J_{\mathbf{x}, \mathbf{x}'}(\mathbf{d})$ , and calculate it directly by solving equations  $\partial J / \partial d_x = 0$  and  $\partial J / \partial d_y = 0$  for  $\mathbf{d}$ . See Appendix for the details.

By this method we can integrate the outputs of arbitrary designs of Gabor filters. Moreover, since  $J_{\mathbf{x}, \mathbf{x}'}(\mathbf{d})$  uses amplitude  $\rho$  for weighting, the influence of unstable phase is attenuated.



**Fig. 1.** Schematics of the use of multiple filters. Each circle represents the coverage of a filter. Left one is a sketch of matching in one shot. Right one shows the iterative matching. The iterative use of filters allows a coverage for wider disparities and a more reliable matching with filters of higher frequencies.

### 3.4 The Choice of Filters

Now that we have the method for integrating multiple filters, we need also consider the choice of filters and how to utilize them in practice. We set the parameters of Gabor filters as a combination of  $\omega_\nu = 2^{-(\nu+1)}\pi$  ( $\nu = 0, \dots, 3$ ) and  $\theta_\mu = \mu\pi/8$  ( $\mu = 0, \dots, 7$ ), and thereby we use  $4 \times 8 = 32$  sorts of Gabor filters. In integrating these filters, we take a coarse-to-fine strategy. Equation (4) tells that  $\mathbf{d}$  obtained from multiple filters is limited to the range  $|\mathbf{d}| \leq \pi/\omega_{\max}$ , where  $\omega_{\max}$  is the greatest central frequency in the group of Gabor filters.

Thus, we group the filters into several sets by their central frequencies,  $\omega_\nu$ , and first use only those with the lowest central frequency. For the obtained matching points, we compute the residual of the distance using the set of filters with the second lowest frequency, and iteratively continue this procedure using filter groups with higher frequencies. In this way, we can use many filters and obtain more reliable matching even when  $\mathbf{d}$  has a large value. Figure 1 is a sketch of this method. In the experiments in Sect. 5, we group 32 filters into four sets and use all of them for matching.

### 3.5 Matching Algorithm

We describe the matching algorithm.  $I^{(k)}$  ( $k = 1, 2, \dots$ ) denotes each frame in a sequence of images.

1. Apply a feature point detector to  $I^{(1)}$ , obtaining feature points  $\mathbf{z}^{(1)} = (z_1^{(1)}, \dots, z_m^{(1)})$ . Set  $k = 1$ .
2. Set points  $\mathbf{z}^{(k+1)'} = (z_1^{(k+1)'}, \dots, z_m^{(k+1)'})$  in  $I^{(k+1)}$ . These points are candidates which is to match  $\mathbf{z}^{(k)}$  or close to  $\mathbf{z}^{(k)}$ . When the difference between  $I^{(k)}$  and  $I^{(k+1)}$  is small, set  $\mathbf{z}^{(k+1)'}$  at the same coordinate as  $\mathbf{z}^{(k)}$  as the initial guess.
3. Match feature points by using (12), obtaining the feature points in  $I^{(k+1)}$ , i.e.  $\mathbf{z}^{(k+1)}$ . By the coarse-to-fine strategy, repeat matching for several times. Set  $k = k + 1$ , and return to step 2.

## 4 Analysis of Phase under Illumination Variances

In this section, we analyze the local phase under illumination variances. We employ the Lambertian reflection model and represent the pixel intensity,  $L(p)$ , of a point  $p$  under incident light by a light source at infinity as

$$L(p) = \mathbf{b}(p) \cdot \mathbf{l}_0 . \quad (13)$$

The 3-vector  $\mathbf{l}_0$  is the product of the strength of the light source with the unit vector for its direction whereas the 3-vector  $\mathbf{b}(p)$  is defined to be the product of the albedo,  $\eta(p)$ , with the inward facing unit normal,  $\hat{\mathbf{b}}(p)$ , for point  $p$ . Hence,  $\mathbf{b}(p) = \eta(p)\hat{\mathbf{b}}(p)$ , and  $\hat{\mathbf{b}}(p)$  can then be considered to encode the 3D shape of the surface around  $p$ , and  $\eta(p)$  to represent the texture of the object.

The positions of detected feature points are according to  $\eta$  and  $\hat{\mathbf{b}}$  since many of feature points detectors, such as Harris corner detector, extract feature points at the position where the image gradients are steep in two dimensions. We can thereby classify feature points into two categories by the parameters which give rise to high gradient. For these two types of feature points, we analyze the phase under illumination variances. We first assume 1D signal  $f(x)$ , which we define as the pixel intensity at position  $x$  as

$$f(x) = \eta(x)\hat{\mathbf{b}}(x) \cdot \mathbf{l}_0 . \quad (14)$$

We then extend the discussions to the case with a 2D signal.

**Feature Points due to Texture.** First, we analyze the group of feature points that are detected due to steep gradient of the texture. We assume that these feature points are only due to texture and  $\hat{\mathbf{b}}(x)$  is constant, i.e.  $\hat{\mathbf{b}}(x) = \hat{\mathbf{b}}_0$ , in the neighborhood of  $x$ . Let  $f'(x)$  be the pixel intensity caused by the incident light  $\mathbf{l}'_0$ . Since the surface normal is constant, the effect of the change from  $\mathbf{l}_0$  to  $\mathbf{l}'_0$  is uniform and the relation between  $f(x)$  and  $f'(x)$  can be written as

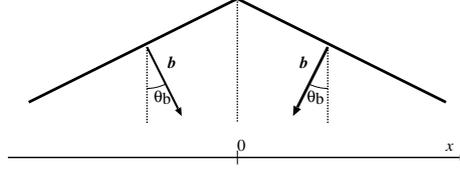
$$f'(x) = \alpha f(x) , \quad (15)$$

where  $\alpha$  is a coefficient. Let  $\rho(\omega)$  and  $\phi(\omega)$  be the amplitude and the phase of  $f(x)$  at frequency  $\omega$ , respectively. Analogously, let  $\rho'(\omega)$  and  $\phi'(\omega)$  be those of  $f'(x)$ . In frequency domain the relation of (15) corresponds to the change of the amplitude, and thus we have

$$\rho'(\omega) = \alpha \rho(\omega) \quad (16)$$

$$\phi'(\omega) = \phi(\omega) . \quad (17)$$

Hence, for 1D signals the local phase of a feature point due to its texture is stable even if the incident light changes. This is also true for 2D signals because the relation in (15) is not limited to the case with 1D signals.



**Fig. 2.** The area where surface normal changes discontinuously.

**Feature Points due to Shape.** Next, we analyze the group of feature points due to the shape of an object. In particular we analyze an surface that is composed of two parts of different normal vector, see Fig. 2. We assume the texture of the parts is constant, i.e.  $\eta(x) = \eta_0$ . Let the coordinate where two different parts are connected be  $x = 0$ , then the normal  $\hat{\mathbf{b}}$  can be expressed as

$$\arg \hat{\mathbf{b}}(x) = \begin{cases} -\theta_b & (x \geq 0) \\ +\theta_b & (x < 0) \end{cases}, \quad (18)$$

where  $\theta_b$  denotes the angle between angle bisector of the corner made by the two different parts and the normal vector of each part. Then the pixel intensity is

$$f(x) = \begin{cases} \eta_0 |\mathbf{l}_0| \cos(\theta_l + \theta_b) & (x \geq 0) \\ \eta_0 |\mathbf{l}_0| \cos(\theta_l - \theta_b) & (x < 0) \end{cases}, \quad (19)$$

where  $\theta_l$  is the angle of  $\mathbf{l}_0$  defined relative to the angle bisector. Equation (19) implies that  $f(x)$  under illumination variance is not uniform. However,  $f(x)$  can be expressed as

$$f(x) = au(x) + b, \quad (20)$$

by using a step function,  $u(x)$ , and coefficients  $a$  and  $b$ . Using this expression, the pixel intensity caused by varying incident light can also be expressed as  $f'(x) = a'u(x) + b'$  with coefficients  $a'$  and  $b'$ .

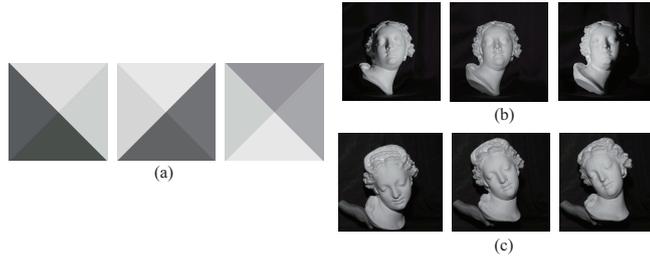
In the frequency domain the counterpart of (20) gives

$$F(\omega) = aU(\omega) + 2\pi b\delta(\omega), \quad (21)$$

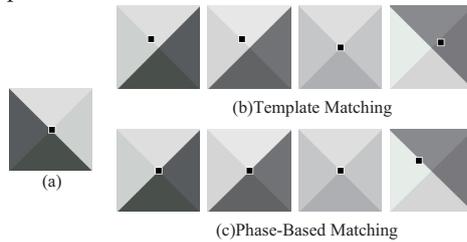
where  $F(\omega)$  and  $U(\omega)$  are the Fourier transforms of  $f(x)$  and  $u(x)$ , respectively. As we have explained in Sect. 2, the DC component is excluded from the Gabor filter. Thus,  $F(\omega) = aU(\omega)$ . The amplitude and the phase change due to the variance of incident light as

$$\rho'(\omega) = \left| \frac{a'}{a} \right| \rho(\omega) \quad (22)$$

$$\phi'(\omega) = \begin{cases} \phi(\omega) & (a'/a \geq 0) \\ -\phi(\omega) & (a'/a < 0) \end{cases}. \quad (23)$$



**Fig. 3.** Examples of images used in the experiments. (a) CG images. (b) Real images of a statue captured under different lighting positions. (c) Real images of the statue captured in different poses.



**Fig. 4.** Matching for CG images. (a) Reference image. (b) Results by template matching. (c) Results by phase-based matching.

This implies that the phase is stable under illumination change as long as  $a'/a \geq 0$ , i.e., as long as the sign of  $\theta_l$  is unchanged.

For 2D signals, we have seen in Sect. 3.1 that the convolution with a 2D Gabor filter is attributed to that of directional function  $\hat{f}_{\theta_0}(\hat{x})$  for direction  $\theta_0$  with 1D Gabor filter. Hence the phase is stable under illumination change as long as  $\hat{f}_{\theta_0}(\hat{x})$  for certain directions satisfy the above described condition and the effect of them are dominant after integrating those functions in (12). In the next section we use CG simulated images to examine the stability of local phases experimentally.

## 5 Experiments

We use three classes of image sequences, including CG images and real images, for evaluating our method. We compare the results with those obtained by conventional template matching – SAD (Sum of Absolute Difference) of mean subtracted pixel intensities between a template and an input image,<sup>2</sup> which reduces the influence of uniform brightness change.

<sup>2</sup> The template size is  $9 \times 9$  pixels and the search area  $21 \times 21$  pixels.

**CG Images.** Images in Fig. 3(a) are generated by viewing a pyramid shaped object with diffuse surface from the top, while changing the position of the light source. Since image gradients are steep in two dimensions at the top of the pyramid, we define a feature point there in the reference image and match it in other images.

Figure 4(a) shows the reference image while (b) and (c) show the results by template matching and phase-based matching, respectively. These results show that the phase-based matching succeeded in some cases where template matching failed. In particular, for the leftmost image of Fig. 4(c), the matching succeeded even though the brightnesses in both side areas have been reversed. As a whole, in this experiment the phase-based matching turned out to be valid when the brightest area is common between two images. This result supports the argument in Sect. 4 that local phase stays constant as long as the change of the light position is limited to a certain range.

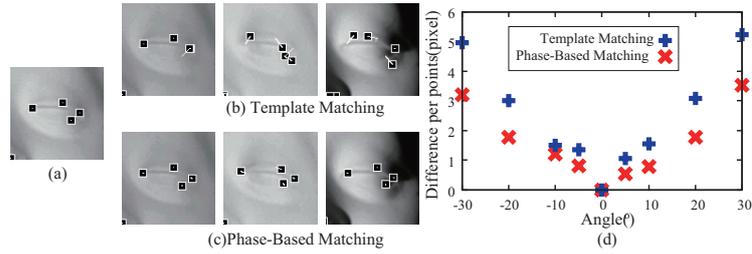
**Real Images under a Moving Light Source.** We search for matching of feature points in real images as shown in Fig. 3(b). These images are captured by viewing a statue from its front, with a light source at  $0^\circ$ ,  $5^\circ$ ,  $10^\circ$ ,  $20^\circ$ ,  $30^\circ$  away from the optic axis of the camera to both left and right sides. We use the image with the light source at  $0^\circ$  as the reference image and detect feature points by Harris corner detector. We then find matching of these points in the other images. Note that we preclude some of them that are shadowed in some images, and those on the boundary of the foreground.

Figure 5 shows the results of matching. Figure 5(d) plots the average displacements of the coordinates of the feature points measured from the reference image for different lighting positions. We can observe that the phase-based matching gives results with less displacements in all cases.

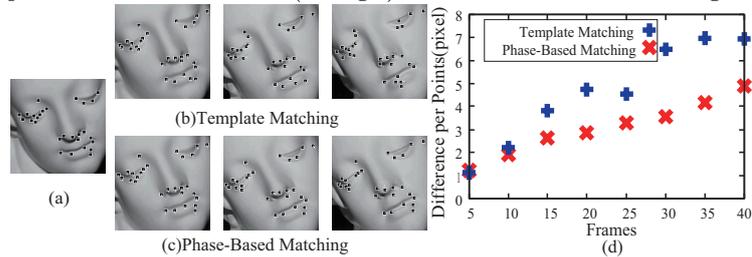
**Real Images of an Object in Motion.** We search for feature points in time series images of a statue which are captured while changing the pose of it as shown in Fig. 3(c). These images are taken at a rate of 30 frames per second and we examine a sequence of 40 images. We extract feature points in the first frame by applying Harris corner detector. Again, we remove those on the boundary of the foreground and those which are occluded in some frames.

For matching feature points between consecutive images, it is natural to refer image  $I^{(k-1)}$  to obtain a corresponding feature point in image  $I^{(k)}$ . In this experiment, however, we choose to determine corresponding feature points in image  $I^{(k)}$  by comparing their local phases or pixel intensities to those in the first frame,  $I^{(1)}$ , so that we can deal with the case that illumination variance under postural change is nontrivial. On the other hand, in order to narrow the search area, we utilize the coordinates of the feature points in  $I^{(k-1)}$  as the initial estimates of those in  $I^{(k)}$ .

To evaluate the performance of the feature matching we use epipolar line. First, we compute the fundamental matrices by using the coordinates of the feature points in the first frame and those of corresponding feature points in



**Fig. 5.** Matching for real images of an object captured under different lighting positions. (a) Reference image. (b) Results by template matching. (c) Results by phase-based matching. (d) Average displacements of feature points (in pixels) for different lighting positions that are defined (in angle) relative to the reference image.



**Fig. 6.** Matching for a sequence of real images of an object in different poses. (a) First frame. (b) Results by template matching. (c) Results by phase-based matching. (d) Average distances (in pixels) between matched points and the epipolar line in the first frame.

each of 5, 10, 15, 20, 25, 30 and 35 frames. Then we use them to draw epipolar lines in the first frame image and check the average distance between matched points and the epipolar line in each case. Since the distance should become zero when the matching is ideally computed, we can employ this value as a measure of accuracy of the feature matching. We assumed affine camera model.

Figure 6 shows the results of matching. Figure 6(d) plots the average distances between matched points and the computed epipolar line. We can observe that the phase-based method realizes smaller average distance in all frames.

## 6 Conclusion

In this paper, we have proposed a phase-based method for feature matching and analyzed its stability under illumination variances. The method turned out to show improved results compared to conventional template matching in situations where illumination changes. In order to discuss the efficiency of our method more generally, we need further theoretical analysis of local phase under lighting changes. Analyses on signals which encode continuous changes in the orientation of objects surfaces would be typical ones. It will also be worth seeking further efficient ways to select and integrate multiple filters.

**Acknowledgements.** We wish to thank Prof. Akihiro Yamamoto for approving the research, and Dr. Takatsugu Hirayama for helpful comments. This work is supported by MEXT, Japan, under a Grant-in-Aid for Scientific Research (No.16680010) and in part by national project on “Development of high fidelity digitization software for large-scale and intangible cultural assets”.

## References

1. Harris, C., Stephens, M.: A combined corner and edge detector. In: Proc. the Fourth Alvey Vision Conference. (1988) 147–152
2. Shi, J., Tomasi, C.: Good features to track. In: Proc. of CVPR’94, IEEE Computer Society (1994) 593–600
3. Smith, S.M., Brady, J.M.: SUSAN – A new approach to low level image processing. (TR95SMS1c) (1995)
4. Fleet, D.J., Jepson, A.D.: Stability of phase information. IEEE Trans. PAMI **15**(12) (1993) 1253–1268
5. Barron, J., Fleet, D.J., Beauchemin, S.: Performance of optical flow techniques. IJCV **12** (1994) 43–77
6. Sanger, T.D.: Stereo disparity computation using gabor filters. Biological Cybernetics **59** (1988) 405–418
7. Maki, A., Bretzner, L., Eklundh, J.O.: Local Fourier phase and disparity estimates: an analytical study. In: Proc. 6th International Conference on Computer Analysis of Images and Patterns. (1995) 868–873
8. Carneiro, G., Jepson, A.D.: Phase-based local features. In: Proc. ECCV’02, Springer-Verlag (2002) 282–296
9. Wiskott, L., Fellous, J.M., Krüger, N., von der Malsburg, C.: Face recognition by elastic bunch graph matching. In: Intelligent Biometric Techniques in Fingerprint and Face Recognition. CRC Press (1999) 355–396
10. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Second edn. Cambridge University Press (2004)

## A Calculation of Distance from Evaluation Function

From (11), we can calculate  $\mathbf{d}$  directly by solving equations  $\partial J/\partial d_x = 0$  and  $\partial J/\partial d_y = 0$  for  $\mathbf{d}$ , as

$$\begin{pmatrix} d_x \\ d_y \end{pmatrix} = \frac{1}{\Gamma_{xx}\Gamma_{yy} - \Gamma_{xy}\Gamma_{yx}} \times \begin{pmatrix} \Gamma_{yy} & -\Gamma_{yx} \\ -\Gamma_{xy} & \Gamma_{xx} \end{pmatrix} \begin{pmatrix} \Phi_x \\ \Phi_y \end{pmatrix}, \quad (24)$$

where  $\Gamma_{xx}\Gamma_{yy} - \Gamma_{xy}\Gamma_{yx}$  must be nonzero, and  $\Gamma_{xy}$ ,  $\Phi_x$  are

$$\Gamma_{xy} = \sum_{j=1}^N \rho_j(\mathbf{x})\rho_j(\mathbf{x}')\omega_{jx}\omega_{jy} \quad (25)$$

$$\Phi_x = \sum_{j=1}^N \rho_j(\mathbf{x})\rho_j(\mathbf{x}')\omega_{jx}\Delta\phi_j(\mathbf{x}, \mathbf{x}'), \quad (26)$$

respectively.  $\Gamma_{xx}$ ,  $\Gamma_{yx}$ ,  $\Gamma_{yy}$  and  $\Phi_y$  are analogously defined by substituting  $\omega_{jx}$  and  $\omega_{jy}$  in (25) and (26).