Blur Insensitive Texture Classification Using Local Phase Quantization

Ville Ojansivu and Janne Heikkilä

Machine Vision Group, Department of Electrical and Information Engineering,
University of Oulu, PO Box 4500, 90014, Finland
{vpo,jth}@ee.oulu.fi

Abstract. In this paper, we propose a new descriptor for texture classification that is robust to image blurring. The descriptor utilizes phase information computed locally in a window for every image position. The phases of the four low-frequency coefficients are decorrelated and uniformly quantized in an eight-dimensional space. A histogram of the resulting code words is created and used as a feature in texture classification. Ideally, the low-frequency phase components are shown to be invariant to centrally symmetric blur. Although this ideal invariance is not completely achieved due to the finite window size, the method is still highly insensitive to blur. Because only phase information is used, the method is also invariant to uniform illumination changes. According to our experiments, the classification accuracy of blurred texture images is much higher with the new method than with the well-known LBP or Gabor filter bank methods. Interestingly, it is also slightly better for textures that are not blurred.

1 Introduction

Natural surfaces usually exhibit some repetitive intensity variations or patterns that are generally referred to as texture. Analysis of texture information is important in machine vision, and it has numerous applications including surface inspection, medical image analysis, and remote sensing [1]. In some applications, image degradations may limit the applicability of the texture information. One class of degradation is blur due to motion, out of focus, or atmospheric turbulence. Because image deblurring is very difficult and introduces new artifacts, it is desirable to be able to analyze texture in a way that is insensitive to blur.

The focus of this paper is on blur insensitive texture classification. There are not many texture analysis methods that are considered to be insensitive to blurring. A blur robust descriptor based on color constancy was proposed in [2]. Also, blur invariant moments [3] or the modified Fourier phase [4] could be used in principle, but they are mainly intended for global object recognition, not local texture analysis.

In this paper, we propose a new blur insensitive texture classification method, which is based on quantized phase of the discrete Fourier transform (DFT) computed in local image windows, and it is called local phase quantization (LPQ).

A. Elmoataz et al. (Eds.): ICISP 2008, LNCS 5099, pp. 236–243, 2008.

The codes produced by the LPQ operator are insensitive to centrally symmetric blur, which includes motion, out of focus, and atmospheric turbulence blur [5]. The LPQ operator is applied to texture identification by computing it locally at every pixel location and presenting the resulting codes as a histogram. Generation of the codes and their histograms is similar to the LBP method [6]. Local frequency analysis, often referred to as signal processing methods, has also been used for texture analysis previously. For a review, see [7]. One of the best known methods uses a bank of Gabor filters and is based on magnitude information [8]. Phase information has been used in [9] and histograms have been used in conjunction with spectral information in [10]. Nevertheless, blur sensitivity has not been considered as a criterion when designing these operators.

First we introduce the conditions under which the DFT phase is invariant to blur in Sect. 2. Then, Sect. 3 proposes the LPQ operator. Section 4 contains experimental results and Sect. 5 presents conclusions.

2 Blur Invariance Using Fourier Transform Phase

In digital image processing, the discrete model for spatially invariant blurring of an original image $f(\mathbf{x})$ resulting in an observed image $g(\mathbf{x})$ can be expressed by a convolution [5], given by

$$g(\mathbf{x}) = (f * h)(\mathbf{x}) \ , \tag{1}$$

where $h(\mathbf{x})$ is the point spread function (PSF) of the blur, $*$ denotes 2-D convolution and $\mathbf{x}$ is a vector of coordinates $[x, y]^T$. In the Fourier domain, this corresponds to

$$G(\mathbf{u}) = F(\mathbf{u}) \cdot H(\mathbf{u}) \ , \tag{2}$$

where $G(\mathbf{u})$, $F(\mathbf{u})$ and $H(\mathbf{u})$ are the discrete Fourier transforms (DFT) of the blurred image $g(\mathbf{x})$, the original image $f(\mathbf{x})$, and the PSF $h(\mathbf{x})$, respectively, and $\mathbf{u}$ is a vector of coordinates $[u, v]^T$. We may separate the magnitude and phase parts of (2), resulting in

$$\begin{aligned} |G(\mathbf{u})| &= |F(\mathbf{u})| \cdot |H(\mathbf{u})| \quad \text{and} \\ \angle G(\mathbf{u}) &= \angle F(\mathbf{u}) + \angle H(\mathbf{u}) \ . \end{aligned} \tag{3}$$

If we assume that the blur PSF $h(\mathbf{x})$ is centrally symmetric, namely $h(\mathbf{x}) = h(-\mathbf{x})$, its Fourier transform is always real-valued, and as a consequence its phase is only a two-valued function, given by

$$\angle H(\mathbf{u}) = \begin{cases} 0 & \text{if } H(\mathbf{u}) \geq 0 \\ \pi & \text{if } H(\mathbf{u}) < 0 \ . \end{cases} \tag{4}$$

This means that

$$\angle G(\mathbf{u}) = \angle F(\mathbf{u}) \quad \text{for all } H(\mathbf{u}) \geq 0 \ . \tag{5}$$

In other words, the phase of the observed image $\angle G(\mathbf{u})$ at the frequencies, where $H(\mathbf{u})$ is positive, is invariant to centrally symmetric blur. In the case of ideal motion and out of focus blur, the cross-section of $h(\mathbf{x})$ is rectangular [5]. This results in a spectrum $H(\mathbf{u})$ of which cross-section is a sinc function containing also negative values. The values of $H(\mathbf{u})$ are always positive before the first zero crossing at frequency $\approx$ (blur length)/(sampling frequency) that satisfies (5). In the case of Gaussian PSF, which models atmospheric turbulence blur [5], $H(\mathbf{u})$ is also Gaussian with only positive values that always satisfy the condition (5).

In practice, blur invariance cannot be completely achieved because of the finite size of the observed images. The convolution of the ideal image with the blur PSF extends beyond the borders of the observed image so that part of the information is lost. When the extent of blur is large enough compared with the image size, this border effect becomes noticeable.

3 Local Phase Quantization for Texture Classification

3.1 Short-Term Fourier Transform

The local phase quantization (LPQ) method is based on the blur invariance property of the Fourier phase spectrum described in Sect. 2. It uses the local phase information extracted using the 2-D DFT or, more precisely, a short-term Fourier transform (STFT) computed over a rectangular M-by-M neighborhood $\mathcal{N}_\mathbf{x}$ at each pixel position $\mathbf{x}$ of the image $f(\mathbf{x})$ defined by

$$F(\mathbf{u},\mathbf{x}) = \sum_{\mathbf{y}\in\mathcal{N}_\mathbf{x}} f(\mathbf{x}-\mathbf{y})e^{-j2\pi\mathbf{u}^T\mathbf{y}} = \mathbf{w}_\mathbf{u}^T\mathbf{f}_\mathbf{x} \ , \tag{6}$$

where $\mathbf{w}_\mathbf{u}$ is the basis vector of the 2-D DFT at frequency $\mathbf{u}$, and $\mathbf{f}_\mathbf{x}$ is another vector containing all M^2 image samples from $\mathcal{N}_\mathbf{x}$.

As it can be noticed from (6), an efficient way of implementing the STFT is to use 2-D convolutions $f(\mathbf{x}) * e^{-2\pi j\mathbf{u}^T\mathbf{x}}$ for all $\mathbf{u}$. Since the basis functions are separable, computation can be performed using 1-D convolutions for the rows and columns successively.

In LPQ only four complex coefficients are considered, corresponding to 2-D frequencies $\mathbf{u}_1 = [a,0]^T$, $\mathbf{u}_2 = [0,a]^T$, $\mathbf{u}_3 = [a,a]^T$, and $\mathbf{u}_4 = [a,-a]^T$, where a is a scalar frequency below the first zero crossing of $H(\mathbf{u})$ that satisfies the condition (5). Let

$$\mathbf{F}_\mathbf{x}^c = [F(\mathbf{u}_1,\mathbf{x}), F(\mathbf{u}_2,\mathbf{x}), F(\mathbf{u}_3,\mathbf{x}), F(\mathbf{u}_4,\mathbf{x})] \ , \quad \text{and} \tag{7}$$

$$\mathbf{F}_\mathbf{x} = [\mathrm{Re}\{\mathbf{F}_\mathbf{x}^c\}, \mathrm{Im}\{\mathbf{F}_\mathbf{x}^c\}]^T \ , \tag{8}$$

where $\mathrm{Re}\{\cdot\}$ and $\mathrm{Im}\{\cdot\}$ return real and imaginary parts of a complex number, respectively. The corresponding 8-by-M^2 transformation matrix is

$$\mathbf{W} = [\mathrm{Re}\{\mathbf{w}_{\mathbf{u}_1}, \mathbf{w}_{\mathbf{u}_2}, \mathbf{w}_{\mathbf{u}_3}, \mathbf{w}_{\mathbf{u}_4}\}, \mathrm{Im}\{\mathbf{w}_{\mathbf{u}_1}, \mathbf{w}_{\mathbf{u}_2}, \mathbf{w}_{\mathbf{u}_3}, \mathbf{w}_{\mathbf{u}_4}\}]^T \ , \tag{9}$$

so that

$$\mathbf{F}_\mathbf{x} = \mathbf{W}\mathbf{f}_\mathbf{x} \ . \tag{10}$$

3.2 Statistical Analysis of the Coefficients

Let us assume that the image function $f(\mathbf{x})$ is a result of a first-order Markov process, where the correlation coefficient between adjacent pixel values is ρ, and the variance of each sample is σ^2. Without a loss of generality we can assume that $\sigma^2 = 1$. As a result, the covariance between positions $\mathbf{x}_i$ and $\mathbf{x}_j$ becomes

$$\sigma_{ij} = \rho^{||\mathbf{x}_i - \mathbf{x}_j||} \ , \tag{11}$$

where $||\cdot||$ denotes L_2 norm, and the covariance matrix of all M samples in $\mathcal{N}_\mathbf{x}$ can be expressed by

$$\mathbf{C} = \begin{bmatrix} 1 & \sigma_{12} & \cdots & \sigma_{1M} \\ \sigma_{21} & 1 & \cdots & \sigma_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{M1} & \sigma_{M2} & \cdots & 1 \end{bmatrix} \ . \tag{12}$$

Hence, the covariance matrix of the transform coefficient vector $\mathbf{F}_\mathbf{x}$ can be obtained from

$$\mathbf{D} = \mathbf{W}\mathbf{C}\mathbf{W}^T \ . \tag{13}$$

One can easily notice that $\mathbf{D}$ is not a diagonal matrix for $\rho > 0$, meaning that the coefficients are correlating.

3.3 Decorrelation and Quantization

Before quantization the coefficients are decorrelated, because it can be shown that the information is maximally preserved in scalar quantization if the samples to be quantized are statistically independent. Assuming Gaussian distribution, independence can be achieved using a whitening transform

$$\mathbf{G}_\mathbf{x} = \mathbf{V}^T\mathbf{F}_\mathbf{x} \ , \tag{14}$$

where $\mathbf{V}$ is an orthonormal matrix derived from the singular value decomposition (SVD) of the matrix $\mathbf{D}$ that is

$$\mathbf{D} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \ . \tag{15}$$

Notice, that $\mathbf{V}$ can be solved in advance for a fixed value of ρ.

Next, $\mathbf{G}_\mathbf{x}$ is computed for all image positions, i.e., $\mathbf{x} \in \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$, and the resulting vectors are quantized using a simple scalar quantizer

$$q_j = \begin{cases} 1\,, \text{ if } g_j \geq 0 \\ 0, \text{ otherwise} \end{cases} \ , \tag{16}$$

where g_j is the jth component of $\mathbf{G}_\mathbf{x}$. The quantized coefficients are represented as integer values between 0-255 using binary coding

$$b = \sum_{j=1}^{8} q_j 2^{j-1} \ . \tag{17}$$

Finally, a histogram of these integer values from all image positions is composed and used as a 256-dimensional feature vector in classification.

The resulting integers b are invariant to centrally symmetric blur provided that the window $\mathcal{N}_{\mathbf{x}}$ is infinitely large and the frequency spectrum of the blur PSF is positive at the sample locations $\mathbf{u}_1 - \mathbf{u}_4$. The second condition is easily met if a is sufficiently small. However, the first condition cannot be fulfilled in practice, and therefore, complete invariance is not achieved, but as shown in the experiments, even a relatively small neighborhood is enough for robustness to reasonable extents of blur.

Decorrelation and quantization do not have any effect on the blur invariance property. In the whitening transform the coefficient vectors are subject to an eight-dimensional rotation that only causes a uniform phase shift to all vectors. In quantization the eight-dimensional space is divided into 256 hypercubes, and the assignment of a vector to one of these hypercubes depends only on the phase information.

4 Experiments

In the experiments, we measured the performance of our LPQ method in the classification of sharp as well as blurred textures. The correlation coefficient was selected to be $\rho = 0.9$ in all the experiments. As test material we used the applicable test suites of the Outex texture image database [1] [11]. For comparison, we also did the same experiments with two other widely known texture classification methods: local binary pattern (LBP) method [2] [12,6] and a method based on Gabor filter banks [3] [8]. We used the Matlab implementations of these reference methods, which can be found on the Internet. Both methods have also been used previously in conjunction with the Outex texture database [11].

All three test suites of the Outex texture database used in our experiments, *Outex_TC_00000-00002*, contained images from 24 texture classes and had 100 different test cases that divided the images into training and test sets differently. Test suites with a larger number are more challenging, as they contain more and smaller images. In the experiments, we used a k-nearest neighbor (k-NN) classifier, which was trained and tested using the appropriate sets of the images. The value of k was 1, 3 or 15 for test suites *Outex_TC_00000-00002*, respectively. We used the Chi square distance measure for the LPQ and LBP histograms. For the Gabor features we used the distance measure proposed in [8]. Notation $\mathrm{LBP}_{P,R}$ means LBP with P samples at radius R. In classical LBP $P = 8$, which results in a code with values in the range {0,...,255}, similar to LPQ. Notation LPQ_R means LPQ of a spatial window with dimensions $M = 2R + 1$. A larger radius R for LPQ and LBP, which provides the comparable spatial extent of the operators, gives better results for blurred textures, but too large radius deteriorates the classification results for sharp textures. The frequency parameter used for LPQ was $a = 1/M$, which is the lowest non-zero frequency.

[1] http://www.outex.oulu.fi/

[2] http://www.ee.oulu.fi/mvg/page/lbp_matlab/

[3] http://vision.ece.ucsb.edu/texture/software/

Table 1. Texture classification accuracies of the non-whitened LPQ, LPQ, LBP and Gabor methods in the first experiment for *Outex_TC_00002* test suite

	LPQ_1 nw	LPQ_1	$LBP_{8,1}$	Gabor
Accuracy	88.0 %	93.6 %	90.2 %	90.2 %

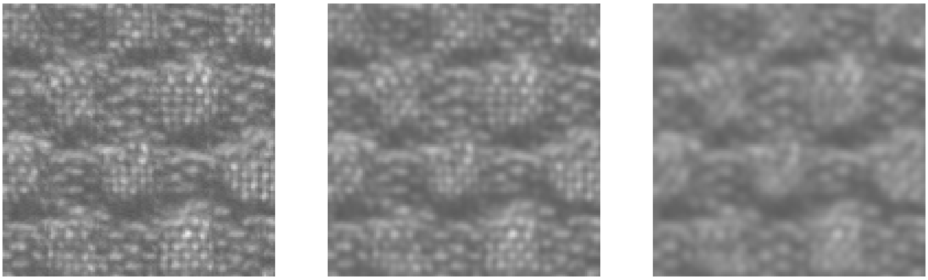
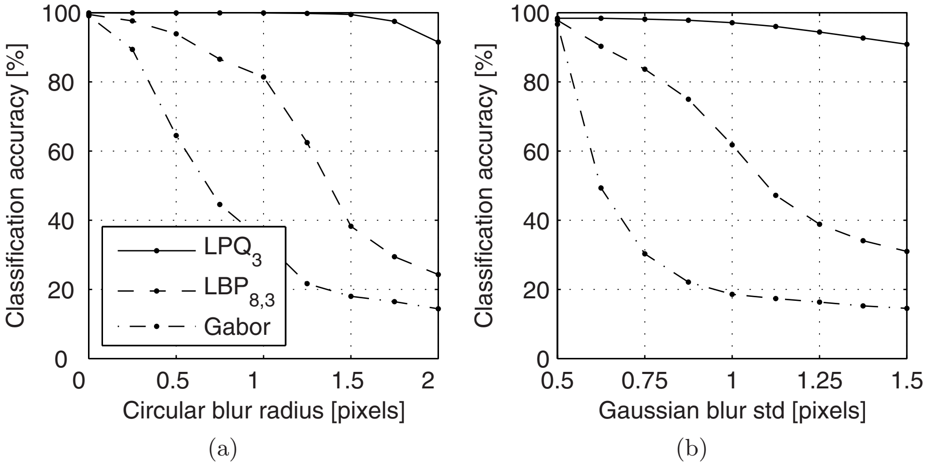


Fig. 1. An example of the texture images used in the second experiment (left). Circularly blurred versions of the same image with blur radii one (middle) and two (right).



(a)

(b)

Fig. 2. Classification accuracies of the LPQ, LBP and Gabor methods for (a) circularly blurred textures of test suite *Outex_TC_00000* and for (b) Gaussian blurred textures of test suite *Outex_TC_00001*

In the first experiment, we tested the classification performance of the methods for the sharp texture images of the challenging *Outex_TC_00002* test suite. The test suite includes 8832 images of size 32 × 32; hence, 368 images per class. We also included the result for non-whitened (nw) LPQ to demonstrate the effect of the whitening transform. We used $LBP_{8,1}$ and LPQ_1, which results in the basic forms of these operators, and the Gabor method. The classification accuracy as percentages is shown in Table 1.

As can be seen, the whitening improves the performance of the LPQ method significantly and the whitened LPQ gives the best score. The success of LPQ also for sharp images was a little surprising, but it was verified also in the other experiments. For some reason the result of the Gabor method for the same test suite was two percentages better in the experiments in [11], where the Gabor method gave the best result. Nevertheless, the result of LPQ is still the best.

In the second and third experiment, we tested the three texture classification methods in the case of blurred textures, which is the main theme of this paper. In the second experiment we used the test suite *Outex_TC_00000*, which includes 480 images of size 128×128, 20 images per class. The classifier was trained using the sharp images, but the test images were artificially blurred using circular and flat PSF, which mimics the out of focus blur [5]. The blur radius was $\{0, 0.25, \ldots, 2\}$. Figure 1 shows three examples of one texture image with blur radii 0, 1, and 2, respectively. We used LPQ and LBP operators with various values of R, but value $R = 3$ seemed to give the best trade-off for different extents of blur. The results for these LPQ_3, $\mathrm{LBP}_{8,3}$, and Gabor methods are shown in Fig. 2(a). As can be seen from the diagram, the LPQ method is very tolerant of blur, while the Gabor method performs worst. Even very small blur deteriorates the results of all but the LPQ methods. We also tried a modification of $\mathrm{LBP}_{8,3}$, namely $\mathrm{LBP}^{u2}_{16,3}$ [6], which uses 16 samples, but the result was only 0.2 percent better for sharp images and worse for blurred images. It is remarkable that the result for LPQ also for sharp images (Blur radius = 0) was the best for any values of R.

In the third experiment, we used the *Outex_TC_00001* test suite, which includes 2112 images of size 64×64 and thus 88 images per class. Now, the artificial blur had Gaussian PSF with standard deviation in the range $\{0.5, 0.75, \ldots, 1.5\}$, which mimics the blur caused, for example, by atmospheric turbulence [5]. Otherwise, the experiment was similar to the second experiment. Again, LPQ_3 and $\mathrm{LBP}_{8,3}$ offered the best trade-off for different extents of blur; therefore, the results of these operators with the Gabor method are shown in Fig. 2(b). The alternative $\mathrm{LBP}^{u2}_{16,3}$ operator did not improve the results. As can be seen, the results of Fig. 2(a) and Fig. 2(b) are quite similar, except that the test suite is a bit more challenging in the latter. The LPQ_3 is again the best option at any blur level, while the Gabor method is the worst. For all values of R the result of LPQ was also the best for the smallest blur (Blur std = 0.5).

5 Conclusions

In this paper, we proposed a new LPQ texture analysis method that operates on the Fourier phase computed locally for a window in every image position. The phases of the four low-frequency coefficients are uniformly quantized into one of 256 hypercubes in eight-dimensional space, which results in an 8-bit code. These LPQ codes for all image pixel neighborhoods are collected into a histogram, which describes the texture and can be used for classification. The phases of the low-frequency components are shown to be ideally invariant to centrally symmetric blur. Although, the invariance is disturbed by the finite-sized image windows, the method is still very tolerant of blur. Because only phase information is used, the method is also invariant to uniform illumination changes.

The proposed method was compared with two well-known texture analysis operators, the LBP and a Gabor filter bank based method. The results of the texture classification experiments on the Outex texture database show that the LPQ method tolerates significantly more blurring than other methods. In addition to that, LPQ also gave slightly better results for sharp texture images.

References

1. Tuceryan, M., Jain, A.K.: Texture analysis. In: Chen, C.H., Pau, L.F., Wang, P.S.P. (eds.) The Handbook of Pattern Recognition and Computer Vision, pp. 207–248. World Scientific Publishing Co, Singapore (1998)
2. van de Weijer, J., Schmid, C.: Blur robust and color constant image decription. In: Proc. IEEE International Conference on Image Processing (ICIP 2006), Atlanta, Georgia, October 2006, pp. 993–996 (2006)
3. Flusser, J., Suk, T.: Degraded image analysis: An invariant approach. IEEE Trans. Pattern Anal. Machine Intell. 20(6), 590–603 (1998)
4. Ojansivu, V., Heikkilä, J.: A method for blur and similarity transform invariant object recognition. In: Proc. International Conference on Image Analysis and Processing (ICIAP 2007), Modena, Italy, September 2007, pp. 583–588 (2007)
5. Banham, M.R., Katsaggelos, A.K.: Digital image restoration. IEEE Signal Processing Mag. 14(2), 24–41 (1997)
6. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. Pattern Anal. Machine Intell. 24(7), 971–987 (2002)
7. Randen, T., Husøy, J.H.: Filtering for texture classification: A comparative study. IEEE Trans. Pattern Anal. Machine Intell. 21(4), 291–310 (1999)
8. Manjunathi, B.S., Ma, W.Y.: Texture features for browsing and retrieval of image data. IEEE Trans. Pattern Anal. Machine Intell. 18(8), 837–842 (1996)
9. Vo, A.P., Oraintara, S., Nguyen, T.T.: Using phase and magnitude information of the complex directional filter bank for texture image retrieval. In: Proc. IEEE International Conference on Image Processing (ICIP 2007), San Antonio, Texas, September 2007, pp. 61–64 (2007)
10. Xiuwen, L., DeLiang, W.: Texture classification using spectral histograms. IEEE Trans. Image Processing 12(6), 661–670 (2003)
11. Ojala, T., Mäenpää, T., Pietikäinen, M., Viertola, J., Kyllönen, J., Huovinen, S.: Outex - new framework for empirical evaluation of texture analysis algorithms. In: Proc. 16th International Conference on Pattern Recognition (ICPR 2002), August 2002, pp. 701–706 (2002)
12. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distribution. Pattern Recognition 29(1), 51–59 (1996)