Multiscale Autoconvolution Histograms for Affine Invariant Pattern Recognition

Esa Rahtu   Mikko Salo   Janne Heikkilä
Department of Electrical and Information Engineering
P.O. Box 4500, 90014 University of Oulu, Finland
{erahtu,msa,jth}@ee.oulu.fi

Abstract

In this paper we present a new way of producing affine invariant histograms from images. The approach is based on a probabilistic interpretation of the image function as in the multiscale autoconvolution (MSA) transform, but the histograms extract much more information of the image than traditional MSA. The new histograms can be considered as generalizations of the image gray scale histogram, encoding also the spatial information. It turns out that the proposed method can be efficiently computed using the Fast Fourier Transform, and it will be shown to have essentially the same computational load as MSA. The experiments performed indicate that the new invariants are capable of reliable classification of complex patterns, outperforming MSA and many other methods.

1   Introduction

Invariant descriptors have drawn much attention in pattern recognition, since they provide an elegant way of dealing with pose changes. These descriptors may have many different levels of invariance, which will determine what kind of pose changes they can cope with. Some approaches have only rotation or scaling invariance, but this is generally not sufficient to model real view angle changes. Instead, the affine transformation has proven to be a reasonably accurate estimate for the pose changes in many occasions.

The affine transformation consists of two components, a translation and a multiplication by a matrix. In many approaches the invariance is achieved by using a combination of normalization and a descriptor with weaker invariance. An extreme example could be normalization of both components and application of a noninvariant descriptor [3]. The problem here is that the accurate normalization of the matrix multiplication is generally difficult for arbitrary shaped patches. In many other methods, like affine invariant moments [4], [8], [9], cross-weighted moments [10], and spatial multiscale affine invariants (SMA) [7], only the effect of the translation is eliminated separately by first translating the image to its centroid. The drawback is that the overall invariant may be strongly affected by the accuracy of finding the centroid, which can be difficult in nonideal conditions. There are also methods like trace transform [5] and affine invariant spectral signatures [2] that are capable of handling the linear transformation and translation together, but they are rather expensive to compute and difficult to implement. Yet another approach could be to use affine invariant Fourier descriptors as proposed in [1]. However this requires extraction of a closed boundary curve of the object.

The multiscale autoconvolution (MSA) transform [6] provides a method for constructing a large number of affine invariants with reasonable computational load and straightforward implementation. The idea behind MSA is to combine the affine coordinate system with a probabilistic interpretation of the image function. One obtains a random variable whose properties are unaffected by the affine transformation, and in [6] the MSA transform was defined to be the mean value of this random variable. However, using only the mean value we capture just a small part of the information carried by this variable. In this paper we propose a histogram method which approximates the whole distribution of this random variable, and the new method turns out to have essentially the same computational complexity as the traditional MSA. The experiments performed indicate that using the new approach we can achieve similar accuracy as in MSA with only a fraction of the transformation parameters.

2 Multiscale autoconvolution

We begin by recalling the ideas behind the multiscale autoconvolution [6] transform, which will then directly lead to the derivation of the new approach. But before continuing we define the spatial affine transformation as follows.

Definition 1 *A spatial affine transformation $\mathscr{A}$ of coordinates $x \in \mathbf{R}^2$ is given by $\mathscr{A}(x) = Tx + t$, where $t, x \in \mathbf{R}^2$ and T is a 2×2 nonsingular matrix whose elements belong to $\mathbf{R}$. Further let $f(x)\colon \mathbf{R}^2 \to \mathbf{R}$, $f \geq 0$ be an image intensity function. The affine transformed version f' of f is given by*

$$f'(x) = f \circ \mathscr{A}^{-1}(x) = f(T^{-1}x - T^{-1}t).$$

2.1 Affine coordinate system

Let $f(x)\colon \mathbf{R}^2 \to \mathbf{R}$, $f \geq 0$ be an image intensity function and let $x_0, x_1, x_2 \in \mathbf{R}^2$ be three points from the support of f. Now given these three points fix a coordinate system using the basis vectors $x_1 - x_0$ and $x_2 - x_0$, and consider a point $u_{\alpha,\beta}$ having the coordinate values $(\alpha, \beta) \in \mathbf{R}^2$ in this system. The point $u_{\alpha,\beta}$ is given in the original coordinate system as

$$u_{\alpha,\beta} = \alpha(x_1 - x_0) + \beta(x_2 - x_0) + x_0.$$

Let $\mathscr{A}$ be an affine transformation applied to the image f as in Definition 1, and take the points $x_0', x_1', x_2' \in \mathbf{R}^2$ from the support of the image $f' = f \circ \mathscr{A}^{-1}$ to be the transformed versions corresponding to x_0, x_1, and x_2, i.e. $x_i' = Tx_i + t$. Also define

$$u_{\alpha,\beta}' = \alpha(x_1' - x_0') + \beta(x_2' - x_0') + x_0'.$$

Using the relation $x_i' = Tx_i + t$ here, we have

$$u_{\alpha,\beta}' = \alpha(Tx_1 - Tx_0) + \beta(Tx_2 - Tx_0) + Tx_0 + t = \mathscr{A}(u_{\alpha,\beta}), \tag{1}$$

indicating that also the point $u_{\alpha,\beta}'$ is the affine image of the point $u_{\alpha,\beta}$. Figure 1 illustrates this phenomenon.

Given a pair of images f and f' we could use several (α, β) pairs and for example the values $f(u_{\alpha,\beta})$ as descriptors to estimate whether the images are the same up to an affine transformation. The problem here is that we would need to find corresponding point triplets x_0, x_1, x_2 and x_0', x_1', x_2' from the images f and f', which can be difficult in general.
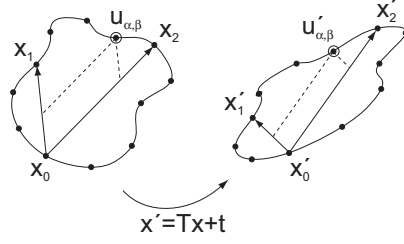
Figure 1: Illustration of the behavior of the affine coordinate system under affine transformation.

2.2 Probabilistic approach

Instead of using fixed points x_0, x_1, and x_2 we assume the points to be independent random variables X_0, X_1, and X_2 with probability density function $p_{X_i}(x) = \frac{1}{\|f\|_{L^1}} f(x)$. Then also $u_{\alpha,\beta}$ will become a random variable

$$U_{\alpha,\beta} = \alpha(X_1 - X_0) + \beta(X_2 - X_0) + X_0.$$

Let now $\mathscr{A}$ be an affine transformation with parameters T and t, and let $f' = f \circ \mathscr{A}^{-1}$ be the transformed image. Using the transformed image, we take X_0', X_1', and X_2' so that they are independent random variables with probability density function $p_{X_i'}(x) = \frac{1}{\|f'\|_{L^1}} f'(x)$. It is not hard to show (see [6]) that then $X_0' \overset{d}{=} \mathscr{A}(X_0)$, $X_1' \overset{d}{=} \mathscr{A}(X_1)$, and $X_2' \overset{d}{=} \mathscr{A}(X_2)$, where we write $X \overset{d}{=} Y$ if the random variables X and Y have the same distribution. Using the X_i' to define $U'_{\alpha,\beta}$ we have as in (1)

$$U'_{\alpha,\beta} = \alpha(X_1' - X_0') + \beta(X_2' - X_0') + X_0' \overset{d}{=} \mathscr{A}(U_{\alpha,\beta}).$$

To eliminate the affine transformation, we consider the random variables $f(U_{\alpha,\beta})$ and $f'(U'_{\alpha,\beta})$, and we get the relation

$$f'(U'_{\alpha,\beta}) = f(\mathscr{A}^{-1}(U'_{\alpha,\beta})) \overset{d}{=} f(\mathscr{A}^{-1}(\mathscr{A}(U_{\alpha,\beta}))) = f(U_{\alpha,\beta}).$$

Hence the random variables $f'(U'_{\alpha,\beta})$ and $f(U_{\alpha,\beta})$ have the same distribution, which is independent of the affine transformation $\mathscr{A}$.

2.3 Definition and expressions for MSA

Since $f'(U'_{\alpha,\beta})$ and $f(U_{\alpha,\beta})$ have the same distribution, they must also have the same expected value. Based on this fact, the multiscale autoconvolution transform was defined in [6] as follows.

Definition 2 (Multiscale autoconvolution) *Let f be a function in $L^1(\mathbf{R}^2) \cap L^2(\mathbf{R}^2)$ with $f \geq 0$, and let $p(x) = \frac{1}{\|f\|_{L^1}} f(x)$ be the corresponding probability density function. Take X_0, X_1 and X_2 to be independent random variables with values in $\mathbf{R}^2$ and with probability density function $p_{X_j}(x_j) = p(x_j)$. For $\alpha, \beta \in \mathbf{R}$ define a new random variable*

$$U_{\alpha,\beta} = X_0 + \alpha(X_1 - X_0) + \beta(X_2 - X_0).$$

The MSA transform of f is defined as the expected value of $f(U_{\alpha,\beta})$,

$$Mf(\alpha,\beta) = E[f(U_{\alpha,\beta})].$$

In [6] it was further noted that given $p(x) = \frac{1}{\|f\|_{L^1}} f(x)$, we can express the probability density function of the variable $U_{\alpha,\beta}$ in Definition 2 as

$$p_{U_{\alpha,\beta}}(u) = (p_\alpha * p_\beta * p_\gamma)(u), \tag{2}$$

where $\gamma = 1 - \alpha - \beta$, $p_a(x) = \frac{1}{a^2} p(\frac{x}{a})$ if $a \neq 0$ and $p_a = \delta(x)$ if $a = 0$ (Dirac delta). Using this result the MSA transform can be expressed in the form

$$Mf(\alpha,\beta) = \int_{\mathbf{R}^2} f(u) p_{U_{\alpha,\beta}}(u) du = \int_{\mathbf{R}^2} f(u)(p_\alpha * p_\beta * p_\gamma)(u) du. \tag{3}$$

Now using (3) we could evaluate the MSA values for a given image function f. However, the double convolution is computationally expensive and thus it was observed in [6] that (3) can be alternatively expressed in terms of the Fourier transform as

$$Mf(\alpha,\beta) = \frac{1}{\hat{f}(0)^3} \int_{\mathbf{R}^2} \hat{f}(-\xi)\hat{f}(\alpha\xi)\hat{f}(\beta\xi)\hat{f}(\gamma\xi) d\xi, \tag{4}$$

where $\hat{f}(\xi) = \mathscr{F}(f) = \int_{\mathbf{R}^2} f(x) e^{-i2\pi x \cdot \xi} dx$ is the Fourier transform of f.

3 The new approach

In the derivation of MSA it was first shown that using an affine coordinate system and a probabilistic approach, we can construct a random variable $f(U_{\alpha,\beta})$ which has the same distribution as $f'(U'_{\alpha,\beta})$ for any affine transformed version f' of f. The descriptor was taken to be the expected value of this random variable. Now it is very likely that by taking only the expected value we will lose much of the information carried by the variable $f(U_{\alpha,\beta})$. In [6] it was remarked that we could also use the higher order moments $E[f(U_{\alpha,\beta})^k]$ as descriptors, but computing each of them contains again another evaluation of an integral similar to (4). In order to capture all the information in $f(U_{\alpha,\beta})$ we would need to estimate the entire distribution of this random variable. Although computing this directly may be difficult, we will show that it can be easily and accurately estimated by a histogram.

To estimate the distribution of $f(U_{\alpha,\beta})$ we would first need the probability density function of $U_{\alpha,\beta}$. Now recall (2), which gives this function in terms of $p(x)$. This expression contains again the computationally expensive double convolution, but by taking Fourier transforms on both sides and noting that the Fourier transform takes convolutions into products, we will have

$$\mathscr{F}(p_{U_{\alpha,\beta}}) = \mathscr{F}(p_\alpha * p_\beta * p_\gamma) = \frac{1}{\hat{f}(0)^3} \hat{f}(\alpha\xi)\hat{f}(\beta\xi)\hat{f}(\gamma\xi). \tag{5}$$

Taking the inverse Fourier transform of (5) we obtain

$$\begin{aligned} p_{U_{\alpha,\beta}}(u) &= \mathscr{F}^{-1}\left(\frac{1}{\hat{f}(0)^3} \hat{f}(\alpha\xi)\hat{f}(\beta\xi)\hat{f}(\gamma\xi)\right) \\ &= \frac{1}{\hat{f}(0)^3} \int_{\mathbf{R}^2} \hat{f}(\alpha\xi)\hat{f}(\beta\xi)\hat{f}(\gamma\xi) e^{i2\pi u \cdot \xi} d\xi. \end{aligned} \tag{6}$$

The expression (6) is very similar to the Fourier form of MSA given in (4), except that here we have $e^{i2\pi u \cdot \xi}$ instead of $\hat{f}(-\xi)$ which actually simplifies the evaluation.

Using (6) we can efficiently calculate the probability density function of $U_{\alpha,\beta}$. Now having this function and the image function $f(x)$, we can perform a simple histogramming operation to establish an estimate of the distribution of $f(U_{\alpha,\beta})$. This is done by dividing the gray level interval into a finite number of disjoint sets B_j (bins) for $j = 1, \ldots, N$. The "mass" in a given bin B_j is the probability that $f(U_{\alpha,\beta})$ lies in B_j, which is given by

$$m_j = P(f(U_{\alpha,\beta}) \in B_j) = P(U_{\alpha,\beta} \in f^{-1}(B_j)) = \int_{f^{-1}(B_j)} p_{U_{\alpha,\beta}}(u)\,du. \qquad (7)$$

The histogram is the vector $(m_1, \ldots, m_N)$, which contains the mass in each bin. If f' is an affine transformed version of f, then since $f(U_{\alpha,\beta})$ and $f'(U'_{\alpha,\beta})$ have the same distribution we have

$$P(f(U_{\alpha,\beta}) \in B_j) = P(f'(U'_{\alpha,\beta}) \in B_j) \quad \text{for each bin } B_j.$$

This shows that the histogram is affine invariant.

For digital images the histogramming requires fewer computations than the corresponding discrete Fourier transform, which makes the computational load of the histogramming approach essentially equal to the evaluation of only the expected value of $f(U_{\alpha,\beta})$. In addition, instead of using the histogram itself as a feature, we can use this simple one dimensional function to produce estimates for descriptors like expectation (MSA), higher order moments, skewness, entropy, etc.

At this point we would like to remark that with parameters $\alpha = \beta = 0$ we obtain exactly the gray value histogram of the image function. In this sense the multiscale autoconvolution histograms can be considered as certain kinds of generalized image histograms which combine the gray value information with the spatial distribution.

4 Implementational issues and computational complexity

When we want to compute the multiscale autoconvolution histograms from digital images, we first need to make a discrete form of the equation (6). However in the MSA case it was found to be better to scale the f rather in spatial domain than in the frequency domain [6]. We do the same here and using the relation $\hat{f}(a\xi) = \hat{f}_a(\xi)$, where $f_a(x) = \frac{1}{a^2} f(\frac{x}{a})$, we can write (6) as

$$p_{U_{\alpha,\beta}}(u) = \mathscr{F}^{-1}\left(\frac{1}{\hat{f}(0)^3} \hat{f}_\alpha(\xi) \hat{f}_\beta(\xi) \hat{f}_\gamma(\xi) \right). \qquad (8)$$

Now if we consider f to be $N_1 \times N_2$ matrix representing the digital image, the approximation of (8) can be done just by replacing the continuous Fourier transforms with the discrete ones. This results in

$$p_{U_{\alpha,\beta}}(u) = \frac{1}{\hat{f}(0)^3} \mathscr{F}^{-1}\left(\hat{f}_\alpha(w) \hat{f}_\beta(w) \hat{f}_\gamma(w) \right), \qquad (9)$$
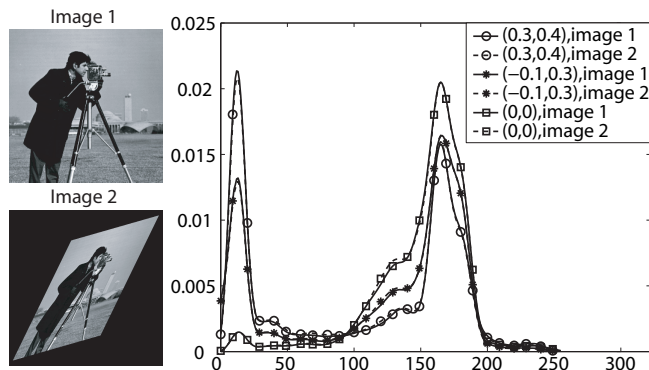
Figure 2: Examples of the multiscale autoconvolution histograms for an image and affine transformed version of it.

where $\hat{f}$ is now a discrete Fourier transform of f and $\mathscr{F}^{-1}$ denotes inverse discrete Fourier transform. This form is quite straightforward to implement, if we just keep in mind that the Fourier transform length must be long enough to achieve the estimation of the linear convolution correctly. If the original image has the size $M_1 \times M_2$, the transform length N_i must satisfy $N_i \geq (|\alpha| + |\beta| + |\gamma|)M_i - 2$.

After we have achieved the estimate of $p_{U_{\alpha,\beta}}(u)$ we have to perform the histogramming operation given in (7). This can be done by first dividing the gray level interval in the desired amount of bins, and then going through all the pixels u in the image f, at each point incrementing the corresponding bin by $p_{U_{\alpha,\beta}}(u)$. The number of histogram bins is a new parameter that can be adjusted, but as in our case the images had discrete gray values in interval $0 \dots 255$ we found it natural to use histogram length 256. Our implementation that we used to perform the experiments in Section 5 can be retrieved from URL: http://www.ee.oulu.fi/research/imag/msa/.

In [6] it was shown that the computational complexity of evaluating one MSA descriptor for an $N \times N$ image is the same as evaluating the Fourier transform, i.e. $O(N^2 log(N))$. Using similar arguments as in the case of MSA we have that in the new approach, the evaluation of $p_{U_{\alpha,\beta}}(u)$ has exactly the same $O(N^2 log(N))$ complexity. The histogram operation, if it is done in the way we described it, has the complexity $O(N^2)$. So the total computational complexity of the histogramming approach is determined by the Fourier transform and is $O(N^2 log(N))$. This means that the multiscale autoconvolution and the proposed multiscale autoconvolution histogram have the same asymptotic complexity.

To have also some insight to the actual computation times, we computed both MSA and the new method using several image sizes N and several α, β pairs. Table 1 illustrates the measured average computation times for one α, β pair with different image sizes using our matlab implementations. From there it can be observed that the computation times of the new approach and traditional MSA are very much in the same range. In addition we

Table 1: Estimated average computation times in seconds for one α, β pair for the proposed method and MSA with different image sizes.

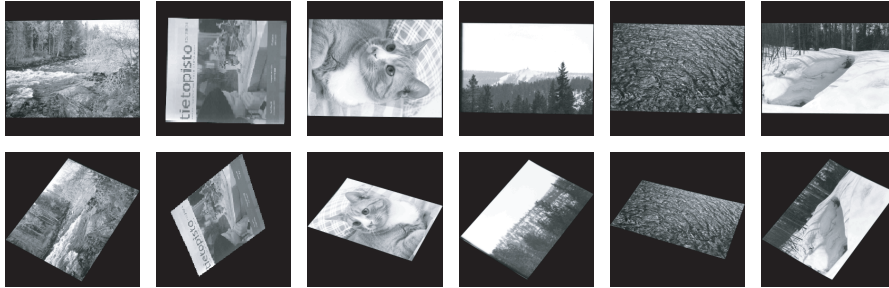| Image size	100×100	200×200	300×300	400×400	500×500
MSA hist.	0.034	0.13	0.33	0.68	0.94
MSA	0.033	0.12	0.29	0.62	0.85

Figure 3: Examples of the training and testing images. The first row contains training images and the second row contains test images.

also checked the accuracy of the histogramming approach by comparing the mean values of the histograms obtained to the corresponding MSA values. As a result we found that the values were exactly the same within the numerical precision, indicating that also the MSA values can be accurately retrieved from the multiscale autoconvolution histograms.

5 Experiments

In this section we present some examples illustrating the behavior of the multiscale autoconvolution histogram compared to multiscale autoconvolution, normalized image gray scale histogram (multiscale autoconvolution histogram with $\alpha = \beta = 0$), and some other commonly used affine invariant techniques. These other techniques were taken to be the spatial multiscale affine (SMA) invariants [7], affine invariant moments (AMI) as described in [4], and affine and photometric moments invariants as described in [9]. As the first example we illustrate multiscale autoconvolution histograms with three α, β pairs in Figure 2. In order to reduce fluctuation in the histograms, we have first slightly smoothed the histogram by convolving it with the Gaussian function having variance 5. Notice the almost exact invariance of the histograms under affine transformation.

In the experiments we use multiscale autoconvolution histogram with $(\alpha, \beta) = (-0.1, 0.3)$, MSA with $(\alpha, \beta) = (-0.1, 0.3)$, and MSA with 10 invariants using $(\alpha, \beta) = \{(-0.1, 0.1), (-0.1, 0.2), (-0.1, 0.3), (-0.1, 0.4), (-0.2, 0.2), (-0.2, 0.3), (-0.2, 0.4), (-0.3, 0.3), (-0.3, 0.4), (-0.4, 0.4)\}$. Since the other comparison methods were faster to evaluate we computed 20 invariants for SMA, 40 invariants for AMI, and all the 15 affine and photometric invariants given in [9]. In addition the affine and photometric invariants described in [9] require a way to divide the object into several affine covariant parts, which we will make similarly as in the experiments in [9]. We will refer to these methods as MSAhist, MSA1, MSA, SMA, AMI, and APMI, respectively. The classification was performed using a simple nearest neighbor classifier, where the distance measure was the histogram intersection for histogram approaches and Mahalanobis distance for the others. The covariance matrix for the Mahalanobis distance was estimated using the features

Table 2: Classification error percents under real view angle distortions.

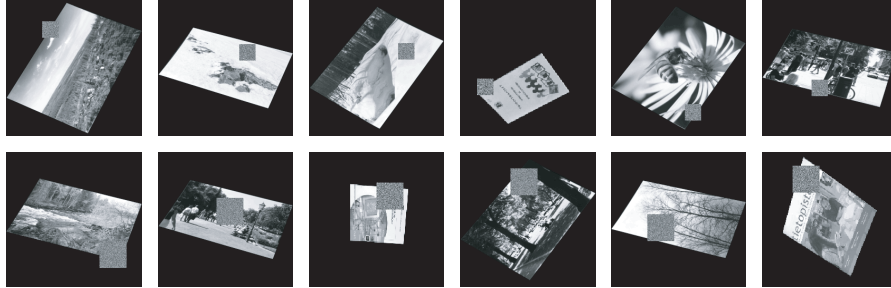| MSA hist.	MSA1	MSA	Gray hist.	SMA	AMI	APMI
2.4 %	81.5 %	13.7 %	12.1 %	29.0 %	50.0 %	69.4 %

Figure 4: Examples of occluded images. At the first row the occlusion has size 30×30 and at the second row 50×50.

from the training samples.

We continue to perform some classification experiments using the procedure described and 45 test objects containing photos, postcards and books. All the test objects were assumed to have rectangular shape in order to make the classification more difficult and allowing us to use the same division method for APMI as in [9]. For training the classifier we took one 256×256 image from each of the 45 test objects. Then for testing we took a new set of photographs containing a total of 124 images from these same objects, where the photographs were taken from various viewing angles. The resulting classification problem is not easy since the test set will also contain many nonaffine distortions due to real photographing conditions and not exactly planar test objects. Before feature extraction the gray values of all the images were normalized so that they had mean 128 and standard deviation 30. This normalization was done in order to diminish the effects of changes in the lighting conditions. This improved the results with all the tested methods except with APMI and thus we applied it to the unnormalized images in the experiments. Figure 3 illustrates some samples of both testing and training images.

Table 2 presents the classification results obtained, where it can be seen that the new approach clearly outperforms the others in classification accuracy. We also observe that the difference between the MSA hist and MSA1 is very large, indicating that there is plenty of information in $f(U_{\alpha,\beta})$ which is completely lost in the expected value. In addition there is a clear difference between the multiscale histograms with $(\alpha, \beta) = (-0.1, 0.3)$ and $(\alpha, \beta) = (0, 0)$, indicating that the spatial information encoded with $(\alpha, \beta) \neq (0, 0)$ can considerably increase the discrimination power of the histograms. The surprise was the poor performance of the APMI. In a separate experiment we found it to be very sensitive to slight changes in the partitioning caused by the nonaffine transformation, which may explain some of the poor performance.

We continue by introducing some additional distortions to the test images in order to

Table 3: Classification error percents under noise distortion with variance v.

v	MSA hist.	MSA	Gray hist.
1	2.4 %	12.9 %	12.1 %
3	4.0 %	13.7 %	12.1 %
5	5.7 %	12.9 %	12.9 %
7	14.5 %	12.9 %	21.7 %
8	17.7 %	15.3 %	21.7 %

Table 4: Classification error percents under illumination distortion with slope s.

s	MSA hist.	MSA	Gray hist.
1 $(\cdot 10^{-4})$	3.2 %	12.9 %	12.1 %
3	4.0 %	13.7 %	10.5 %
5	7.2 %	14.5 %	20.6 %
8	16.1 %	14.5 %	37.9 %

Table 5: Classification error percents under occlusions of size $d \times d$.

Ocl. size	MSA hist.	MSA	Gray hist.
10×10	2.4 %	14.5 %	12.1 %
20×20	2.4 %	15.3 %	12.1 %
30×30	2.4 %	19.4 %	12.9 %
40×40	7.3 %	31.5 %	12.9 %
50×50	15.3 %	39.5 %	19.4 %

see how the methods react to them. As distortions we use Gaussian noise, nonuniform illumination, and image occlusion. The noise was additive and it was generated from a normal distribution with zero mean and variance v. The nonuniform illumination was generated so that starting from the vertical center line, to the left we linearly decreased the gray-scale values and to the right we similarly increased them. The slope of the illumination change is denoted as s. Finally, the occlusion was made by introducing randomly located square patches of size $d \times d$ on the objects. The patches contained normally distributed gray values with mean 128 and variance 20. Some of these occluded images are illustrated in Figure 4. Tables 3, 4 and 5 show the corresponding classification results with different types and levels of distortion. We omitted the MSA1, SMA, AMI, and APMI methods from these experiments due to their high error rates in the first experiment.

The results in the distortion experiment illustrate the differences in the nature of the methods. In the case of Gaussian noise and nonuniform illumination we can observe that histogram approaches react to the changes more quickly than MSA. This is probably due to the fact that the distortions are spread quite evenly over the whole histogram, and the intersection distances are affected more than the mean value of the histogram computed in MSA. The situation can be changed by introducing more smoothing over the histograms. In this case we observed that the multiscale autoconvolution histograms perform much better under noise, but then the results with pure view angle distortions drop slightly. The amount of smoothing can be thought as tradeoff between robustness and discriminability of the method. The effect of occlusions is much better handled by the histograms, since they affect only a small part of the histogram but change the mean value considerably. The results indicate that MSA histograms can actually tolerate considerably large occlusions.

6 Conclusions

In this paper we introduce a new way of constructing affine invariant histograms from images. The proposed approach uses the same affine invariant random variable as the multiscale autoconvolution, but it is capable of estimating the complete distribution rather than just the expected value. It was further shown that the computational complexity for the new method and for MSA was essentially the same. The experiments performed illustrated the performance of the proposed approach and support the fact that there is plenty of discriminating information in $f(U_{\alpha,\beta})$ which is not captured in the expected

value given by MSA. The approach introduced was also shown to clearly outperform many of the commonly used affine invariant techniques. We performed the experiments using just a simple setting and only one (α, β) pair, but we expect that by taking more pairs and combining their information with sophisticated methods like adaboosting or support vector machines, we could achieve still better performance and discriminability. The efficient use of several histograms together and the selection of the best (α, β) pairs are important future research issues.

7   Acknowledgments

References

[1] K. Arbter, W. E. Snyder, H. Burkhardt, and G. Hirzinger. Application of affine-invariant Fourier descriptors to recognition of 3-D objects. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12(7):640–647, July 1990.

[2] J. Ben-Arie and Z. Wang. Pictorial recognition of objects employing affine invariance in the frequency domain. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 20(6):604–618, June 1998.

[3] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, October 2005.

[4] J. Flusser and T. Suk. Pattern recognition by affine moment invariants. *Pattern Recognition*, 26(1):167–174, January 1993.

[5] M. Petrou and A. Kadyrov. Affine invariant features from the trace transform. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 26(1):30–44, January 2004.

[6] E. Rahtu, M. Salo, and J. Heikkilä. Affine invariant pattern recognition using multi-scale autoconvolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(6):908–918, June 2005.

[7] E. Rahtu, M. Salo, and J. Heikkilä. A new efficient method for producing global affine invariants. In *Proc. International Conference on Image Analysis and Processing*, pages 407–414, Cagliari, Italy, 2005.

[8] T. H. Reiss. The revised fundamental theorem of moment invariants. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(8):830–834, August 1991.

[9] L. Van Gool, T. Moons, and D. Ungureanu. Geometric/photometric invariants for planar intensity patterns. In *Proc. European Conference on Computer Vision*, pages 642–651, 1996.

[10] Z. Yang and F. S. Cohen. Cross-weighted moments and affine invariants for image registration and matching. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 21(8):804–814, August 1999.