A New Efficient Method for Producing Global Affine Invariants

Esa Rahtu[1], Mikko Salo[2], and Janne Heikkilä[1]

[1] Machine Vision Group
Department of Electrical and Information Engineering
P.O.Box 4500, 90014 University of Oulu, Finland
{erahtu,jth}@ee.oulu.fi
[2] Rolf Nevanlinna Institute
Department of Mathematics and Statistics
P.O. Box 68, 00014 University of Helsinki, Finland
msa@rni.helsinki.fi

Abstract. This paper introduces a new efficient way for computing affine invariant features from gray-scale images. The method is based on a novel image transform which produces infinitely many different invariants, and is applicable directly to isolated image patches without further segmentation. Among methods in this class only the affine invariant moments have as low complexity as our method, but as known they also possess many considerable weaknesses, including sensitivity to noise and occlusions. According to performed experiments it turns out that our novel method is more robust against these nonaffine distortions observed in image acquisition process, and even in practice its computation time is equivalent to that of the affine invariant moments. It is also observed that already a small subset of these new features is enough for successful classification.

1   Introduction

Recognizing the contents of images has long been one of the main topics in computer vision research. This task is very challenging since already the results of the image acquisition process are affected by many factors making the actual recognition process even more complicated. One of the key problems rising during the imaging process are the geometric transformations caused by the distortions in the pose of the objects. A sophisticated solution is to find a set of descriptors invariant to these deformations, which then provide preferable inputs for classification and recognition algorithms.

Different types of invariants have been under research already for quite some time, but not so many of the introduced methods are capable of producing invariants in complex cases like affine and projective transformations. However, these are often just the models we need to give adequate approximations for real photographing situations. In addition most of the methods capable of handling these transformations are based on salient points or contours found using some other, possibly error prone, techniques.

Global affine invariant methods, capable of producing affine invariant features computed directly from gray-scale images, include affine invariant moments (AIM) [1, 2], cross-weighted moments [3], Ben-Arie's frequency domain technique [4], trace transform [5], and Multi-Scale Autoconvolution (MSA) [6, 7]. Many of these introduce advanced ideas and give high accuracy, but usually at the expense of high complexity and computational demands. This makes them impractical in systems with very limited amount of processing capacity, not to speak of real time systems. Only the affine invariant moments make an exception, and in fact in terms of computational speed AIMs are clearly superior to any other method in this class. However they do have considerable weaknesses, including sensitivity to nonaffine deformations like noise, occlusions, and nonlinearities, making the use of this method in many applications impossible.

In this paper we propose a new method for producing global affine invariants which has the same complexity as affine invariant moments. The proposed novel image transform immediately gives infinitely many different invariants, and according to experiments it turns out to be more robust with respect to noise and occlusions as affine invariant moments. Our method has some similarities with recently proposed Multi-Scale Autoconvolution, but is significantly faster to compute.

2   Motivation

We begin with a short discussion of the Multi-Scale Autoconvolution (MSA) transform [6], which will motivate a new affine invariant transform in Section 3. Let $f : \mathbf{R}^2 \to \mathbf{R}$ be an image function. The Fourier form of MSA is

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

where $\hat{f}(\xi) = \int_{\mathbf{R}^2} e^{-2\pi i x \cdot \xi} f(x) \, dx$ and $\gamma = 1 - \alpha - \beta$.

This form produces affine invariants: $M(f \circ \mathscr{A}^{-1}) = Mf$ for any affine transformation $\mathscr{A}$. This is due to two facts. If f is replaced by the translation $g(x) = f(x - x_0)$, then $\hat{g}(\xi) = e^{-2\pi i x_0 \cdot \xi} \hat{f}(\xi)$, and the choice $\alpha + \beta + \gamma = 1$ makes the exponentials cancel in (1). Also, if f is replaced by $g(x) = f(A^{-1}x)$ where A is a 2×2 nonsingular matrix, then $\hat{g}(\xi) = |\det A|\hat{f}(A^t\xi)$, i.e. $\hat{g}$ is obtained from $\hat{f}$ essentially by composing with a matrix. If one then makes the change of variables $\xi \mapsto A^{-t}\xi$ in (1) everything cancels and we see that (1) is also invariant under composition with nonsingular matrices.

If f represents an $N \times N$ image then the computational complexity of evaluating (1) is essentially the same as for computing the FFT of f, which is $O(N^2 \log N)$. Although this is fairly good compared to many global affine invariant methods, it is still much slower than the $O(N^2)$ complexity of affine invariant moments.

3 Definition

We base our new method on the formula (1), but instead of using the Fourier transform we just use the function f. This will give an expression which is invariant under composition with matrices, but one loses translation invariance. To deal with this we will normalize f by computing the centroid

$$\mu(f) = \frac{1}{\int_{\mathbf{R}^2} f(x)\,dx} \left(\int_{\mathbf{R}^2} x_1 f(x)\,dx, \int_{\mathbf{R}^2} x_2 f(x)\,dx \right)^t$$

and using the normalized function $\tilde{f}(x) = f(x + \mu(f))$.

Definition. *Let $f \in L^\infty(\mathbf{R}^2)$ be compactly supported. For $\alpha, \beta \in \mathbf{R}$ define*

$$If(\alpha, \beta) = \frac{1}{\|f\|_{L^1}} \int_{\mathbf{R}^2} \tilde{f}(x)\tilde{f}(\alpha x)\tilde{f}(\beta x)\,dx. \tag{2}$$

Proposition 1. *$I(f \circ \mathscr{A}^{-1}) = If$ for any affine transformation $\mathscr{A}$.*

Proof. We first show that I is translation invariant. If $g(x) = f(x - x_0)$ then $\|g\|_{L^1} = \|f\|_{L^1}$ and $\mu(g) = \mu(f) + x_0$. Thus $\tilde{g}(x) = g(x + \mu(g)) = \tilde{f}(x)$, and consequently $Ig = If$. Also, if A is a nonsingular matrix let $g(x) = f(A^{-1}x)$. Since $\mu(g) = A\mu(f)$ one has $\tilde{g}(x) = f(A^{-1}(x + A\mu(f))) = \tilde{f}(A^{-1}x)$ and

$$Ig(\alpha, \beta) = \frac{1}{|\det A|\|f\|_{L^1}} \int_{\mathbf{R}^2} \tilde{f}(A^{-1}x)\tilde{f}(\alpha A^{-1}x)\tilde{f}(\beta A^{-1}x)\,dx.$$

The change of variables $x \mapsto Ax$ gives that $Ig = If$. $\qquad\square$

Remark 1. One obtains more general global affine invariants in the following way. Let $H : \mathbf{R}^k \to \mathbf{R}$ be a measurable function, and define

$$I'f(\alpha_1, \ldots, \alpha_k) = \frac{1}{\|f\|_{L^1}} \int_{\mathbf{R}^2} H(\tilde{f}(\alpha_1 x), \ldots, \tilde{f}(\alpha_k x))\,dx. \tag{3}$$

If H satisfies some conditions (e.g. H continuous and $H(\bar{0}) = 0$), then the expression is well defined for compactly supported $f \in L^\infty(\mathbf{R}^2)$.

The proof of Proposition 1 shows that I' is also affine invariant. By changing variables $x \mapsto \alpha_1^{-1}x$ we see that no information is lost if we normalize $\alpha_1 = 1$. In (2) we chose $k = 3$ and $H(x, y, z) = xyz$ and made the normalization $\alpha_1 = 1$. This choice seems natural and it gave good results in the experiments, however we expect that other choices for H may be useful in affine invariant classification.

The transform $f \mapsto If$ has the following symmetries, which are easily obtained from (2) by changes of variables.

Proposition 2. (a) $If(\alpha, \beta) = If(\beta, \alpha)$,
(b) $If(\alpha, \beta) = \alpha^{-2}If(1/\alpha, \beta/\alpha)$ *for* $\alpha \neq 0$,
(c) $If(\alpha, \beta) = \beta^{-2}If(1/\beta, \alpha/\beta)$ *for* $\beta \neq 0$.

The symmetries may be used to show that it is enough to compute $If(\alpha, \beta)$ in the triangle $T = \text{convex hull}\,\{(-1, -1), (-1, 1), (1, 1)\}$. The values $If(\alpha, \beta)$ outside T may be computed from the values in T using the symmetries.

In Figure 1 we have illustrated two sample images and logarithms of their transforms in range $\alpha, \beta \in [-1, 1]$. Notice the symmetries of the transform.

4 Completeness issues

According to our knowledge none of the global affine invariant methods have yet been proved to be complete, i.e. that knowing all the invariant values would uniquely determine the image up to affine transformation. Despite this fact, in practice many of the proposed methods perform fine with natural images. There is some nonuniqueness with our new method, and we will discuss this next.

If we consider binary images which are starlike and symmetric with respect to their centroids, one can see that the values $If(\alpha, \beta)$ are the same for all these sets (a set K is starlike if for any $x \in K$ the line segment between the centroid and x lies in K). Another case arises if f and g are two images with centroid at the origin such that $f(x) = g(-x)$ in C and $f(x) = g(x)$ outside of C, where C is a two-sided sector (i.e. $tx \in C$ whenever $x \in C$ and $t \in \mathbf{R}$). Then $If = Ig$ even though the images may not be related by affine transformation.

The reference method, affine invariant moments, has a similar weakness if only low order moments are considered. If g is any $2M$ times continuously differentiable compactly supported function, it is easy to see that the functions f and $f + \partial_{x_1}^M \partial_{x_2}^M g$ have the same moments up to order $M - 1$. This gives infinitely many images which are not related by affine transformation but which are not distinguished by moments up to order $M - 1$. Usually the moments used in practice are only up to order 3.

It could be argued that in practice images which cause mentioned nonuniqueness arise rarely, and in many applications one may be able to use $If(\alpha, \beta)$ for successful affine invariant classification. However, in applications where classification accuracy is more important than speed, more sophisticated methods like Multi-Scale Autoconvolution may give better results.

5 Implementation and computational complexity

In this section we consider some issues in the implementation of $If(\alpha, \beta)$ and show that it has $O(N^2)$ complexity. The Matlab program we used for computing $If(\alpha, \beta)$ in our experiments in Section 6 is also available at website: http://www.ee.oulu.fi/research/imag/msa/.

To apply the method to digital images one needs a discrete form of the invariant, and this is done by discretizing the integral (2) resulting in

$$If(\alpha, \beta) = \frac{1}{\sum_{x,y} f(x,y)} \sum_{x,y} \tilde{f}(x,y) \tilde{f}(\alpha x, \alpha y) \tilde{f}(\beta x, \beta y). \qquad (4)$$

To evaluate this we need a method to get $\tilde{f}(x,y)$ from the original image function $f(x,y)$. Recall that $\tilde{f}(x,y)$ is a translated version of original image $f(x,y)$ so that the origin of the image coordinates (x, y) coincides with the image centroid. The coordinates of the centroid however rarely are integers, and if we just translate the pixel values to their new coordinates the resulting image does not have its values in the integer grid anymore. In (4) we need samples on three grids
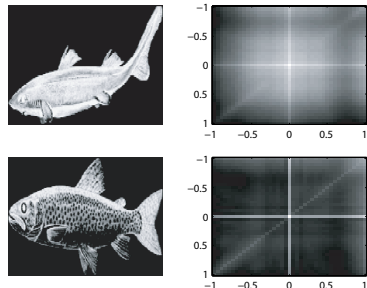
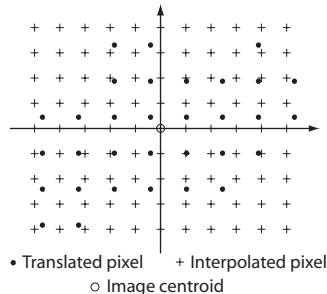**Fig. 1.** Two images and logarithms of their transforms.

Fig. 2. Example of interpolation scheme.

centered at the origin and having sample intervals $\{1, \alpha, \beta\}$, and thus we need to interpolate them from the translated image. This is illustrated in Figure 2. It should be noted that one may select the interpolation grid so that it will have samples needed for all used (α, β) pairs, and we can achieve all desired invariants with only one interpolation. In our implementation we used bilinear interpolation as it seemed to perform well. After having the interpolated values the calculation of the resulting invariants is quite straightforward.

The symmetries in Proposition 2 indicate that we would only need to take (α, β)s from the triangle $\{(-1, -1), (-1, 1), (1, 1)\}$, and thus we chose to take uniform sampling with interval 0.25. However we also believe that already a smaller set would be enough for classification, and to test this we chose a subset of 10 pairs out of these 36. We will denote this invariant set by S10. In addition, if the chosen (α, β) pairs have integer values, the needed interpolation would be minimal reducing the computational demands. Hence we also took two sets of integer (α, β)s, one having 5 and other 10 pairs, addressed as IS10 and IS5 respectively. The three selected (α, β) sets are shown in Figure 3.

By looking at (4) one can directly see that the computational complexity is $O(N^2)$ for an $N \times N$ image. In addition finding $\tilde{f}(x, y)$ requires also interpolation to $N \times N$ points having $O(N^2)$ complexity, which results in an overall complexity $O(N^2)$. Furthermore it should be noted that with this same interpolation we are
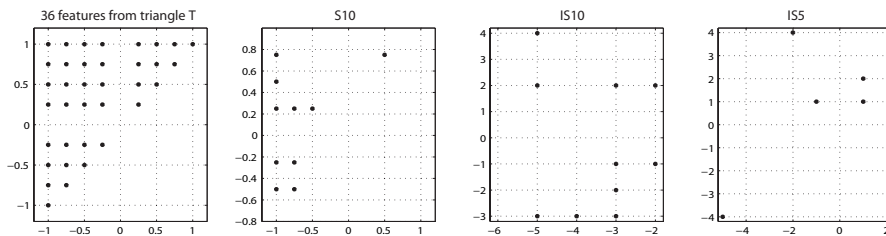

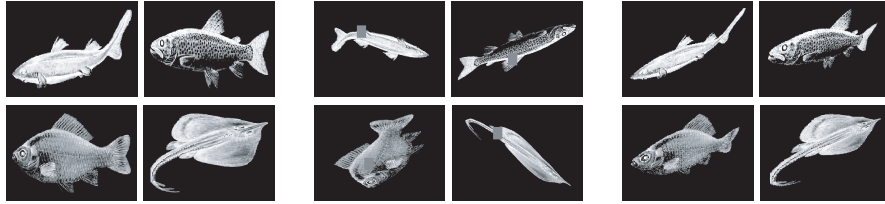**Fig. 3.** (α, β) pairs used in the experiments.

Fig. 4. Samples of fish images used in the experiments. On left there are 4 original images, in the middle 4 occluded images with occlusion size 30 and on the right 4 projective transformed images with nonlinearity value $2 \cdot 10^{-3}$.

able to get also all other invariants that have parameters $(a \cdot \alpha, b \cdot \beta)$ where $a, b \in \mathbb{N}$. For affine invariant moments it quite straightforward to see that the resulting complexity is the same $O(N^2)$ for an $N \times N$ image.

6   Experiments

In this section we perform some experiments assessing our new method, implemented as described in Section 5. We compare the achieved results with affine invariant moments with 4, 10 and 60 independent invariants achieved using polynomials given in [2]. We will refer to these methods as AIM4, AIM10 and AIM60, respectively. For the experiment we took 94 gray scale images of fish, resolution 200×400, and used them to train a nearest neighbor classifier for each tested method. Then the classification performance was estimated using these same images disturbed by a random affine transformation combined with one of the following nonaffine distortions, Gaussian noise, occlusion, and nonuniform illumination. We also tested the performance in the presence of projective transformation. We now define how these distortions were created and illustrate some examples of these and the original images in Figure 4.

The Gaussian noise was uniformly distributed to whole image and then using thresholding removed from the background to eliminate the effect of the background size. The occlusion was generated by introducing different sized and randomly situated square shapes on the objects. The illumination distortion was done 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 projective transformation is defined by the expression

$$\mathscr{P}(x) = \left(\frac{p_{11}x_1 + p_{12}x_2 + p_{13}}{p_{31}x_1 + p_{32}x_2 + p_{33}}, \frac{p_{21}x_1 + p_{22}x_2 + p_{23}}{p_{31}x_1 + p_{32}x_2 + p_{33}} \right),$$

where $p_{31}x_1 + p_{32}x_2 + p_{33} \neq 0$. In our experiment these parameters were chosen so that $p_{11} = p_{22} = p_{33} = 1$, $p_{12} = p_{13} = p_{21} = p_{23} = 0$, and the value $c = |p_{31}| + |p_{32}|$, which can be seen as a measure for the nonlinearity of the transformation, was varied.

To reduce the effect of randomness we performed each test, excluding projective transformation test, 1880 times and took the mean errors of these runs to be the resulting estimates. We have plotted these resulting error rates in Figure 5.
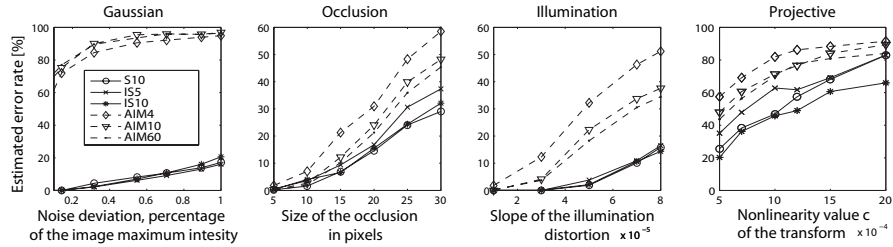
Fig. 5. The estimated error rates in different experiments with all tested methods.

From there it can be observed that our new method performs well under affine transformations and it seems to give better results than affine invariant moments in all experiments. Especially in the Gaussian noise and nonuniform illumination tests the difference is very clear. A significant improvement can also be seen in the results of projective transformation test, thought this kind of distortion seems to be difficult to handle for the both of the tested methods. If we take a look at the different forms of our method, it can be noted that there are slight performance differences in some tests. In most cases the trend is that using more features we get slightly better results, especially in the projective transformation and occlusion tests. However this is not always the case and already 5 invariants give reasonable results. Thus one should select such features that best meet the overall requirements of the application. In addition we also believe that for small images one should not use very large (α, β), since then the scaled images would have only few nonzero pixels.

Comparing the affine invariant moments together, excluding the Gaussian noise test we get better results if we use more invariants, but the improvement seems to get quite small when we have already 10 invariants. In the Gaussian noise test the trend was to the opposite direction, but one can argue that the errors with any number of invariants are so high that the method is useless in such classification tasks anyway.

For final confirmation we performed an experiment to see how the actual computing times relate with these methods. We also added the cross-weighted moments with four invariants and MSA with 30 invariants to the same test to have some reference. We took an image having all pixels nonzero and measured the computation time for each method as a function of image size. In Figure 6 we have two plots, the first with results only of AIM, S10 and IS5, and the second showing also two comparison methods. From these results it can be observed that our new method has also in practice the same computational complexity as affine invariant moments, depending a bit on which (α, β) pairs are used. However as shown by the classification experiments already the IS5 has better performance than AIMs. The results also illustrate how these two methods are clearly in their own category as far as computation time is concerned and this might be very crucial from the perspective of practical applications. We further note that these computations were made using Matlab and one can get even better performance using more optimized implementations with e.g. C language.
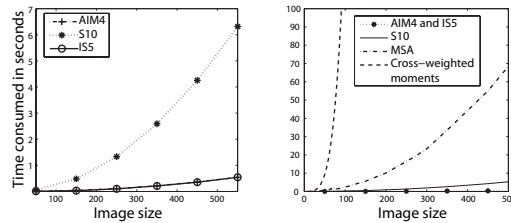
Fig. 6. The measured computation times for AIM4, S10, IS5, 4 cross-weighted moments and 30 MSA values.

7 Conclusions

In this paper we have presented a novel efficient affine invariant image transform, which can provide a good alternative for affine invariant moments in applications where speed and low complexity are crucial. We assessed the new method in image classification tasks where it seemed to clearly outperform affine invariant moments, and it was observed that in many cases a small subset of all features is enough for successful classification. We expect that as the functionality of many mobile and small-scale systems increases, there is greater demand for fast and accurate image analysis methods. Thus we believe that our novel method can offer new possibilities for such applications.

Acknowledgments

References

1. J. Flusser and T. Suk, "Pattern recognition by affine moment invariants," *Pattern Recognition*, vol. 26, no. 1, pp. 167–174, 1993.
2. T. Suk and J. Flusser, "Graph method for generating affine moment invariants," *Proc. International Conference on Pattern Recognition*, vol. 2, pp. 192–195, 2004.
3. Z. Yang and F. Cohen, "Cross-weighted moments and affine invariants for image registration and matching," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 8, pp. 804–814, 1999.
4. J. Ben-Arie and Z. Wang, "Pictorial recognition of objects employing affine invariance in the frequency domain," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 6, pp. 604–618, 1998.
5. M. Petrou and A. Kadyrov, "Affine invariant features from the trace transform," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 26, no. 1, pp. 30–44, 2004.
6. J. Heikkilä, "Multi-scale autoconvolution for affine invariant pattern recognition," in *Proc. International Conference on Pattern Recognition*, pp. 119–122, 2002.
7. E. Rahtu and J. Heikkilä, "Object classification with multi-scale autoconvolution," in *Proc. International Conference on Pattern Recognition*, vol. 3, pp. 37-40, 2004.