

Affine Symmetry and Applications in Image Processing

Supervised By
Graham R. Martin

Heechan Park

Department of Computer Science

University of Warwick

February 8, 2007

Abstract

Natural images have a great deal of self-similarity in a sense that a part of an image can replace another part with a slight deformation. We employ affine symmetry to address the self-relationship, i.e. rotation, scaling, shearing and translation. In this report, various applications are presented exploiting affine symmetry in images. Firstly, an image coding technique is discussed that uses affine symmetric redundancy between blocks. This poses an interesting approach as opposed to the recent directional Wavelet based trend. Experiments show visually acceptable picture quality at low bitrates. Secondly, a segmentation algorithm is presented that analyses images in a similar manner to the image coding application. The algorithm uses a directional shape extracted from the local Fourier spectrum. From the obtained shape, a feature is extracted using the affine-invariant Fourier descriptor. Experimental evaluation on structural textures shows encouraging results and application on natural images demonstrates identification of texture objects. Thirdly, a denoising technique that combines Independent Component Analysis (ICA) and the Multiresolution Fourier Transform (MFT) is presented. This technique inherits the ability to find bases adaptively from given data using ICA and the computational efficiency of the MFT. Another denoising approach is demonstrated that utilises the shape information from the segmentation algorithm as a thresholding mechanism. The experimental results are promising even compared with the recent directional transform such as the Curvelet. Lastly, future research directions are briefly mentioned.

Table of Contents

1	Image Coding	1
1.1	Introduction	1
1.2	Two-Component Analysis	2
1.3	Classification and Prototype Selection	3
1.3.1	Independent Component Analysis (ICA)	4
1.3.2	Classification	5
1.4	Structural Transform Estimation	6
1.4.1	Alternative Method	7
1.5	Parameter Entropy Coding Strategy	10
1.5.1	Zero-Basis Region	10
1.5.2	Vector Quantisation	10
1.6	Experimental Evaluation	11
1.7	Conclusion	12
	Bibliography	15
2	Image Denoising	17
2.1	Introduction	17
2.2	ICA Based Approach	19
2.3	MFT Based Approach	21
2.4	ICA-MFT Combined Approach	21
2.5	MFT-Radon Approach	22
2.6	Experimental Results	26
2.7	Conclusion	26
	Bibliography	28
3	Image Segmentation	29
3.1	Introduction	29

3.2	Affine Invariant Features	31
3.2.1	Fourier Transform Spectrum	31
3.2.2	Directional Shape Radon Analysis	31
3.2.3	Affine Invariant Shape Descriptor	33
3.2.4	Convex Hull - Area Ratio	34
3.3	Affine Invariant Classification	34
3.4	Experiments	35
3.5	Conclusion	37
	Bibliography	38
4	Future work	39
4.1	Image Coding	39
4.2	Image Denoising	39
4.3	Image Segmentation	40
4.4	Video: Motion Estimation	40
4.5	Video: Texture Synthesis	41
	Bibliography	42

List of Figures

1.1	Affine Symmetric Texture	2
1.2	ICA overcomplete basis functions	4
1.3	ICA Block Classification	5
1.4	Classification and Prototype	6
1.5	Fingerprint Reconstruction Test	8
1.6	Magnitude Modeling Using EM	9
1.7	Affine Symmetric Texture Synthesis Experiment	14
2.1	Radon-MFT Denoising	22
2.2	Denoising Test : Lena	24
2.3	Denoising Test : Jaguar	25
3.1	Directional Shape Information Extraction	32
3.2	Segmentation Test on Sample Texture Composites	33
3.3	Segmentation on Natural Images	37

List of Tables

1.1	Image Coding Experimental Result	12
2.1	Denoising Result in SNR	23

Chapter 1

Image Coding

1.1 Introduction

Natural images contain considerable redundancy and this is exploited by state of the art wavelet image compression techniques(1). A recent tendency is to decompose the image into various directional subbands(2). In this context, exploiting the affine symmetric redundancy residing between blocks, namely translation, scaling and rotation, offers a different view-point from which to tackle the compression task. This approach resembles fractal theory as only a single prototypical block is needed to replicate other blocks of similar texture. Hsu and Wilson developed a two-component analysis(3), a framework that combines a structural component which identifies the affine transformation from block to block as shown in Fig.1.1 and a stochastic residual component. The technique provides high fidelity in the reconstruction of periodic texture. Recently, the framework was extended for affine invariant segmentation of natural images by Bhalerao and Wilson(4). Without loss of generality, a natural image comprises regions of different textural pattern. Classification of blocks into self-similar groups and selection of the most representative blocks have paramount impact on the visual quality of reconstruction. They showed that a clustering approach is appropriate and performs well for a supervised case(4). We propose a practical algorithm

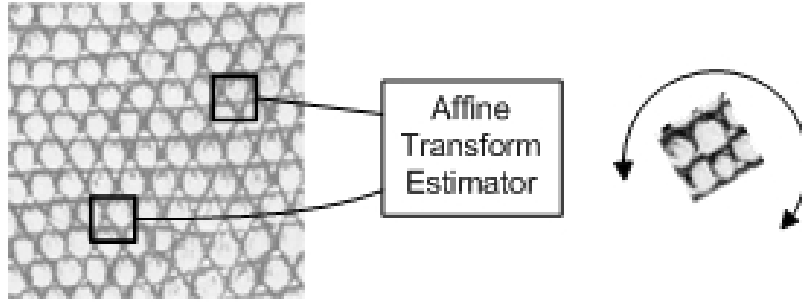


Figure 1.1: Affine symmetric texture synthesis

based on Independent Component Analysis(ICA) for an unsupervised case, resulting in near-optimal prototypical blocks. We also present an alternative structural transformation estimation technique with minimal computational complexity and a parameter entropy coding strategy. In section 1.2, we briefly review two-component analysis before introducing classification of blocks in section 1.3. In section 1.4, we explore structure transform estimation and discuss an alternative approach. In section 1.5, an efficient parameter coding strategy is described. Finally, experimental results are reported and conclusions are drawn.

1.2 Two-Component Analysis



The two-component framework(3) combines a structural element based on the 2D affine relationships between texture patches and a stochastic element designed to account for the unpredictable variations in the texture. We focus on only the structural component. Given an optimal prototypical block f_i and a region of self-similarity composed of affine symmetric blocks, a target patch, f_j is estimated as

$$\hat{f}_j(y) = \alpha_{ij} f_i(T_{ij}(y)) \quad T_{ij}(y) = A_{ij}y + t_{ij} \quad (1.1)$$

$$\alpha_{ij}^2 = \Sigma f_i^2(y) / \Sigma f_j^2(y) \quad (1.2)$$

where $y = (p, q)^T, 0 \leq p, q \leq N$, T_{ij} is a 2D affine transformation of the coordinates of prototype f_i , and α_{ij} is an intensity scaling coefficient. Fig.1.1 illustrates the concept. The Fourier transform is used to reduce the search space by separating the linear and translation estimations which relate to amplitude and phase respectively. For the linear part, A_{ij} , a two energy cluster model is employed to represent the spectra using angular variance analysis. This is conducted to discover angles, θ_1 and θ_2 , which minimise the sum of the variances, σ^2 . Readers are referred to (3) for details.

$$\sigma^2(\theta_1, \theta_2) = \sum_{x \in \Lambda} |x| \cdot \|x - \mu\|^2 \quad (1.3)$$

where μ is the centroid of a cluster and Λ is a set of coordinates in the subdivision formed by θ_1 and θ_2 . The corresponding affine transform is found by aligning the centroids of a cluster in the prototype and the target. Translation, t_{ij} is shown as a phase gradient which is found from the location of the peak in the cross-correlation.

$$\max(F^{-1}f_i(x) \cdot \tilde{f}_j(x)) \quad \tilde{f} = T_{ij}(f_j(x)) \quad (1.4)$$

The resulting blocks are stitched together using a squared cosine mask, W , of size $B \times B$,

$$W(p, q) = \cos^2[\pi p/2B] \cos^2[\pi q/2B] \quad (1.5)$$

1.3 Classification and Prototype Selection

Classification and the identification of appropriate prototypical blocks is an essential step to exploiting affine-symmetry in our approach. The most obvious procedure is to perform an exhaustive search. That is, to compute the transformation error for all combinations of prototype and target and then to choose those that minimise the overall error. This leads to the discovery of the optimal prototype block but is clearly a computationally arduous

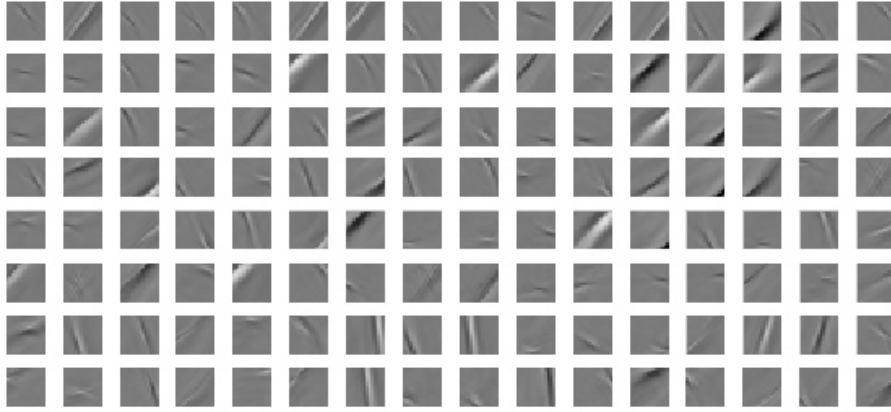


Figure 1.2: ICA overcomplete basis functions

task given the number of combinations. The computational burden can be halved under the symmetry assumption:

$$\forall x, y \quad |f(x, y) - f(y, x)| \cong 0 \quad (1.6)$$

where $f(x, y)$ transforms x to fit y . Nonetheless, the computational requirement becomes prohibitive as the image size increases.

1.3.1 Independent Component Analysis (ICA)

ICA and its variants have been proposed to provide representations that utilise a set of linear basis functions(5; 6). What the approaches have in common is that they try to reduce the information redundancy by capturing the statistical structure in the images, beyond second order information. ICA finds a linear non-orthogonal coordinate system in multivariate data, determined by the higher-order statistics. ICA produces an overcomplete set of basis functions, as shown in Fig.1.2, which is well localised in frequency and orientation. This allows one to find common structure between blocks as well as how similar they are to each other.

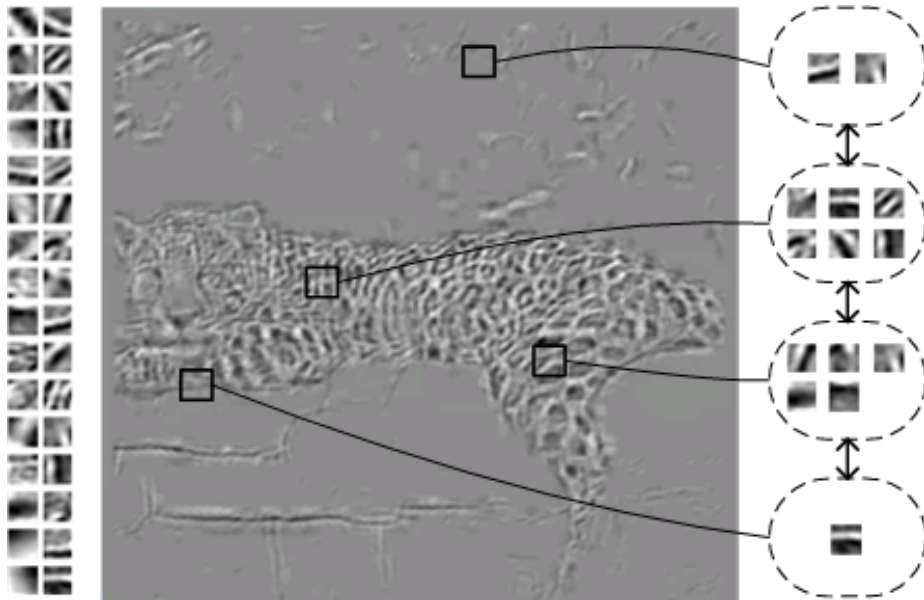


Figure 1.3: Classification: a coarse reconstruction formed from a limited number of ICA bases and a different number of bases for different blocks

1.3.2 Classification

Despite the fact that ICA identifies common directional linear structures, it is not trivial to estimate affine invariance between blocks with only the discovered basis functions. However, the number of the directional linear bases in each block captured by ICA is affine invariant feature as shown in Fig.1.3. This makes sense as it is unlikely to fit one to another if the number of directional linear features does not match. The different number of significant basis functions for each block can be found as

$$\frac{\sum_{i \in s}^k w_i}{\sum w} \leq c \quad 0 \leq k \leq k_{max} \quad (1.7)$$

where w_i is the weight of b_i , c is a constant, and s is a set of basis function indices sorted by $w_i / \| b_i \|$ where b_i is the basis function. k_{max} is the maximum number of basis functions for a block. Blocks are grouped based on the number of significant basis functions in each block. Blocks in each group are compared, if they are not affine invariant, the group is

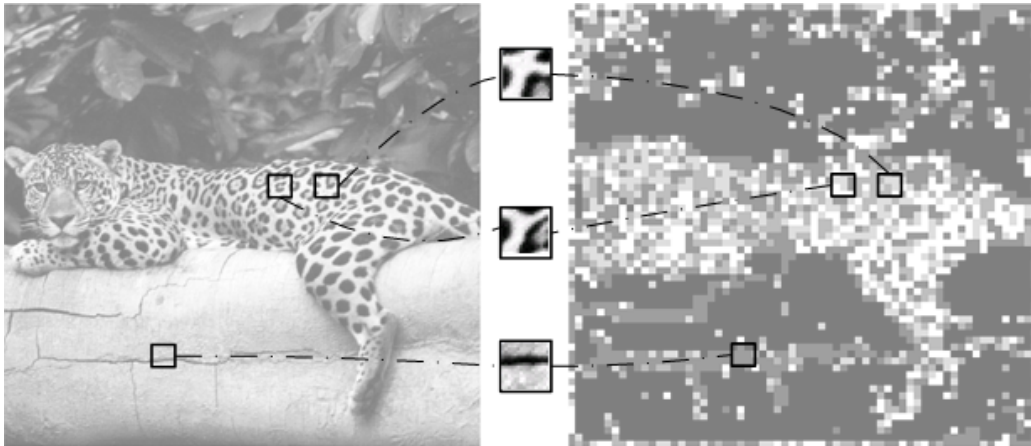


Figure 1.4: Selected prototypes and classification map sub-divided. This can prevent different texture blocks with the same number of directional features from being in the same group. A classification map is shown in Fig.1.4. For each group, a prototypical block is given by

$$\max\{\Sigma(|w_i|/e_2)\} \quad (1.8)$$

where e_2 is the eigenvalue of the 2nd principal component. This results in a block composed of strong directional bases with high weights.

1.4 Structural Transform Estimation



The two-component analysis accurately estimates the texture's structure. However, it is often the case that a two-cluster model produces an incorrect estimate particularly for directional texture. Calway addressed this problem using a simple metric(7). It is assumed that two clusters are present when $\sigma_{1+2}^2/\sigma_{min}^2 < 1$ holds. However, the metric provides a false prediction in some cases, as shown in Fig.1.5. Bhalerao and Wilson developed an alternative strategy to overcome this problem, utilising the Gaussian mixture model(GMM)(8; 4). The

amplitude spectrum is modeled as a two-dimensional, two component Gaussian mixture and approximated using the Levenberg-Marquardt(LM) algorithm. The LM is a non-linear optimisation and requires that the partial derivative of each parameter is known. With the estimated Gaussian mixture model, the LM is applied again to find an optimal affine transformation applied to the Gaussian mixture that best fits the amplitude spectrum of the target block in a least-square error sense. Although this strategy overcomes the fore-mentioned problem, it increases the computational requirement by introducing another step to fit the model to the spectrum.

1.4.1 Alternative Method



Expectation Maximisation for GMM

Expectation Maximisation (EM)(9) is an estimation algorithm driven by each data points' probability of membership to a particular class. The EM is a favourite in statistics due to its simplicity and fast convergence rate, but is apt to fall into local optima. Estimation is achieved by repeating two steps, an E-step and M-step. An example follows with respect to the Gaussian mixture model. The E-step computes expected classes for all data points for each class and is given by

$$P(\omega_i|x_k, \mu_t, c_t) = \frac{p(x_k|\omega_i, \mu_i, c_i)\omega_i}{\sum_{j=1}^c p(x_k|\omega_j, \mu_j, c_j)\omega_j} \quad (1.9)$$

where μ , c and ω are the centre, covarian matrix and weight of each Gaussian model. In the M-step, the maximum likelihood is estimated given the new membership distribution for the data

$$\mu_i = \frac{\sum_k P(\omega_i|x_k, \mu_t, c_t)x_k}{\sum_k P(\omega_i|x_k, \mu_t, c_t)} \quad (1.10)$$



Figure 1.5: Finger print image test: (left)original image,(middle)reconstruction by two-component method, (right) by modified EM for GMM

$$c_i = \frac{\sum_k P(\omega_i | x_k, \mu_t, c_t) (x_k - \mu_i)(x_k - \mu_i)^T}{\sum_k P(\omega_i | x_k, \mu_t, c_t)} \quad (1.11)$$

$$\omega_i = \frac{\sum_k P(\omega_i | x_k, \mu_t, c_t)}{\#data_point} \quad (1.12)$$

Combination of GMM and Two-Component Analysis

The clustering problem in the two-component analysis arose from assuming two clusters model in a single cluster spectrum(3), leading to two centroids μ_i , sit too close. These centroids result in severe deformation when aligned to centroids based on two-cluster data. A method to determine the effective number of clusters and which does not increase the computational requirement is a useful step forward. Our work adopts the GMM estimation of(4) and the alignment of centroids from (3). The LM is substituted by a modified EM, in which the principle component of the Gaussian model is utilised in favour of controlling its orientation and breadth. The Gaussian mixture model is used not just to model the amplitude spectrum but each constituent Gaussian model represents a different directional feature.

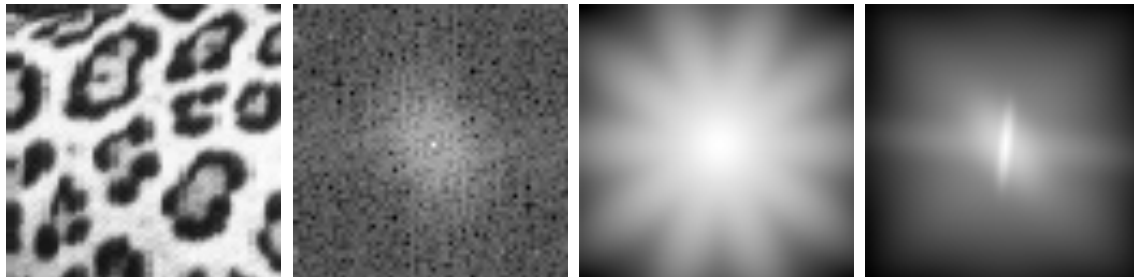


Figure 1.6: Magnitude modeling using EM : (left to right) image block, magnitude spectrum, gaussian mixture before EM iteration, and approximation after iteration

Modified EM for GMM

The Gaussian mixture is initialised as having constituent Gaussian models evenly distributed between 0 and π to avoid local minima. This is done by assigning a designated vector as the principal component of each model. As shown in Fig.1.6, a six directional Gaussian mixture model is used initially to approximate the spectrum. One component is fixed as being non-directional

$$\text{covMatrix}(0,0) = \text{covMatrix}(1,1) \quad (1.13)$$

with $\omega_i = 0.4$. This is to assure the directional components focus on significant features without being distracted by the scattered high frequency data. As iteration progresses, components are fixed as additional background components when their degree of directionality, $e_2/e_1 < 0.6$, or their weights are negligible ($\omega_i < 0.05$), and are excluded in the M-step. As it converges, components left non-fixed are assumed to represent different directional features. These in turn provide an affine transformation by aligning the principal components of two significant Gaussian models. If more than two Gaussian models are left non-fixed, then the model with the least weight, ω_i , and the one with the closest Euclidean L_2 distance to its principal vector X_i are merged as their median. Further iterations are performed until the number of models becomes equal to the number of components of the mixture model of the prototypical block.

1.5 Parameter Entropy Coding Strategy

1.5.1 Zero-Basis Region

Using a single prototype to represent a set of self-similar blocks sounds efficient. However, each structural transformation from prototype to target block entails six affine parameters, a costly representation, particularly for a homogeneous block. The simpler the structure of the block's contents, the less efficient it is to use the structural transform, in terms of bitrate. It is required to distinguish blocks that are better represented by the structural transform from those that are not. We have already solved this problem from the ICA result obtained in Fig.1.4. Blocks containing only non-directional or lowpass filtered like bases as given below are replicated by intensity scaling only, for which Eq.(2) is modified as

$$\alpha_{ij} = E[y] \quad \text{if} \quad \left| \begin{array}{l} \|b_i\| \geq b_{thre} \\ e_2 \geq e_{2thre} \end{array} \right| \quad (1.14)$$

where E is the mean of the block, and α_{ij} is applied on a unit intensity block instead of the prototype.

1.5.2 Vector Quantisation

Another issue making compression of the transform parameters difficult is that they are floating point numbers and sensitive to loss of significant digits when limited precision is applied. We encode centroid coordinates instead of affine parameters. Limited precision can be flexibly applied depending on the probability of the centroids. For a 16×16 block, a grid is given depending on the distance, d from the centre.

$$\begin{array}{l} \text{if } (0 < d < 1 || 7 < d < 8) \text{ apply double-pel} \\ \text{else if } (2 < d < 4) \text{ apply half-pel} \\ \text{else apply full-pel} \end{array} \quad (1.15)$$

Each centroid coordinate pair can be converted into a single byte when a 16×16 block is considered. Therefore, a transform parameter results in three bytes including offset coordinates, prior to first-order entropy coding. First-order entropy coding can be used as, due to the symmetric property of the Fourier transform, centroid neighbourhood redundancy is less likely to exist.

1.6 Experimental Evaluation

The algorithm was executed on the lowest two levels of the Laplacian pyramid. A block entails six parameters for the affine transform, an intensity scale and a prototype id. They are separately encoded using arithmetic coding. An image of size $W \times H$ requires $2W/B \times 2H/B$ blocks of size $B \times B$, the low-pass image of the Laplacian pyramid of size $W/4 \times H/4$ and N prototypes of size $B \times B$. The algorithm was tested on several images of size 512×512 and a block size of 16×16 . The results are shown in Table.1.1, where each column represents the algorithm with different options. A and B are with prototype selection by ICA, and C and D are with manually selected prototypes. A and C are with a classification map by ICA and B and D are with application of all prototypes to the whole image. E is JPEG2000. The top five rows are with three prototypes and the bottom five rows are with two prototypes. Manually selected prototypes may not be optimal choices, however these were the best combinations of those tested. The images are better reconstructed with manually selected prototypes than with ICA based prototypes but the difference is only 0.1dB. We believe that the differences with the optimal prototype based reconstruction would be insignificant. The more prototypes the better the reconstruction but jaguar and giraffe in A and B show the opposite effect. This is due to the weak-classification and sub-optimal prototypes. Comparing A with B and C with D demonstrates that the Zero-Basis Region is an effective strategy to reduce bitrate and that classification by the number of directional features is effective. In general, the rate distortion performance does not match

Table 1.1: Experimental Results [PSNR and bitrate(bpp)]

/#prototype	A	B	C	D	E
	25.18	25.35	25.25	25.45	30.18
barbara/3	0.35	0.52	0.43	0.52	0.35
	23.56	23.62	23.76	23.81	27.38
frog/3	0.43	0.49	0.37	0.50	0.43
	27.25	27.40	27.25	27.39	32.75
giraffe/3	0.35	0.50	0.33	0.50	0.35
	22.64	22.79	22.84	23.04	28.07
jaguar/3	0.34	0.51	0.37	0.52	0.34
	29.17	29.26	29.21	29.33	34.52
lena/3	0.28	0.50	0.37	0.50	0.28
	23.44	23.45	25.50	25.73	27.31
barbara/2	0.21	0.39	0.25	0.39	0.21
	23.36	23.40	23.69	23.74	25.98
frog/2	0.27	0.37	0.21	0.37	0.27
	27.31	27.44	27.20	27.33	30.94
giraffe/2	0.22	0.38	0.20	0.38	0.22
	22.67	22.83	22.74	22.93	26.25
jaguar/2	0.21	0.38	0.21	0.38	0.21
	29.17	29.25	29.22	29.32	31.55
lena/2	0.14	0.38	0.26	0.38	0.14

that of JPEG2000, however our technique provides a satisfying result in that high frequency information is preserved. This is demonstrated by the reconstruction of A and D shown in Fig.1.7.

1.7 Conclusion

An alternative image compression technique exploiting the affine symmetric redundancy present in natural images has been discussed. The main difficulty lies in the selection of prototypical blocks as natural images are often composed of various textures. The technique involves classification of blocks into self-similar groups as well as the selection of an optimal block in each group. A near-optimal solution is suggested that performs ICA based

classification by directional features and which selects the most representative block as the prototype. The reconstruction quality differs by only 0.1dB from those reconstructed by manually selected prototypes. A computationally efficient structural component estimation technique is presented, with an adaptive accuracy integerisation strategy. The work may be useful in applications where detailed texture is preferred to blurred images at low bitrates.

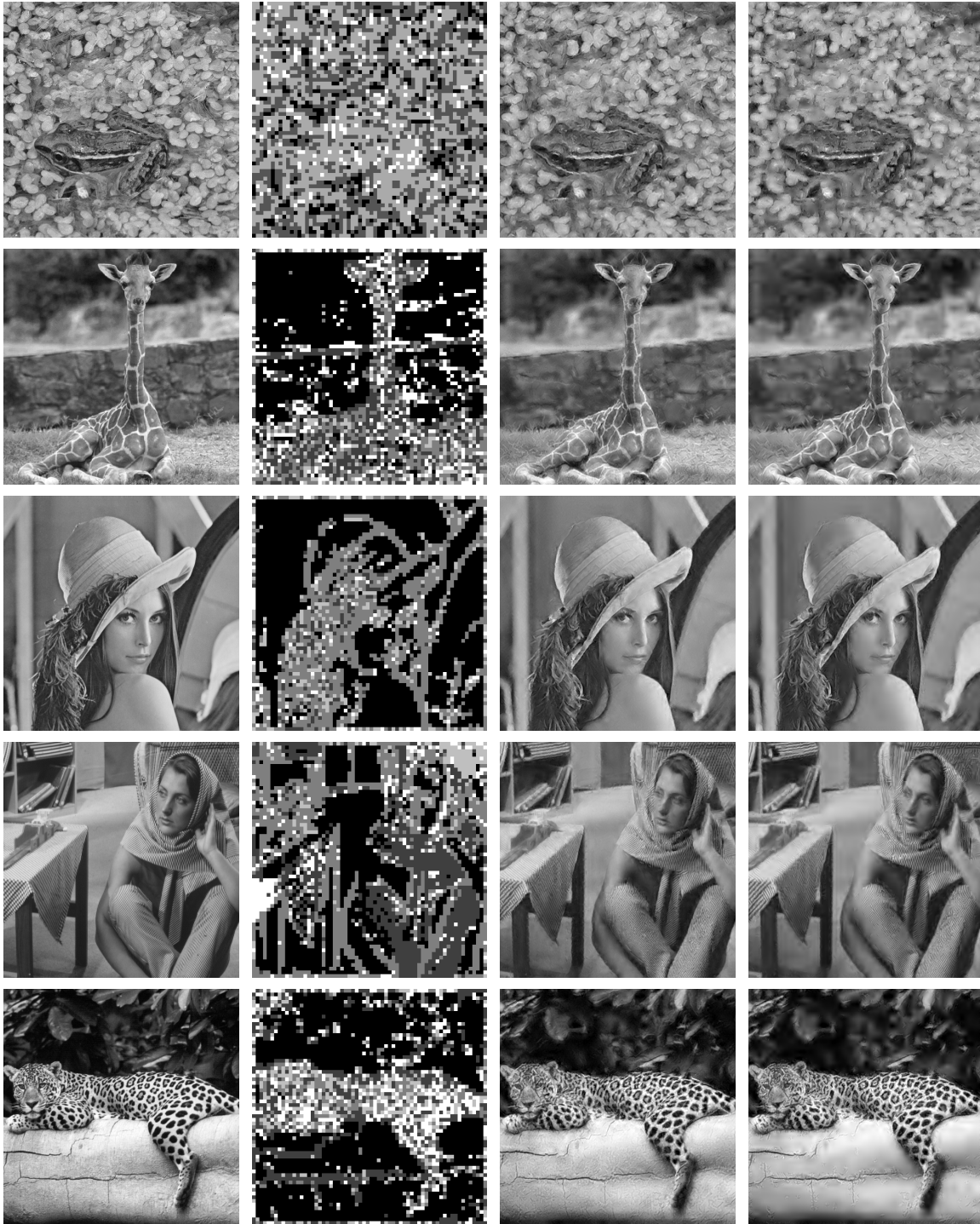


Figure 1.7: Affine symmetric texture synthesis results with three prototypes (from top to bottom) frog, giraffe, lena, barbara, and jaguar (left to right) original image, classification map, reconstruction by applying all prototypes to the whole image(D) and map based reconstruction(A)

Bibliography

- [1] S.D. Taubman and M.W. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards and Practice*, Kluwer Academic Publishers, 2002.
- [2] J.L. Starck, E.J. Candes, and D.L. Donoho, “The curvelet transform for image denoising,” *IEEE Trans. on Image Processing*, vol. 11, pp. 670–684, 2000.
- [3] T.I. Hsu and R. Wilson, “A two-component model of texture for analysis and synthesis,” *IEEE Trans. on Image Processing*, vol. 7, no. 10, pp. 1466–1476, October 1998.
- [4] A. Bhalerao and R. Wilson, “Affine invariant image segmentation,” in *BMVC.*, 2004.
- [5] A. Hyvarinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, John Wiley, 2001.
- [6] B.A. Olshausen, “Learning sparse, overcomplete representations of time-varying natural images,” in *IEEE ICIP.*, 2003, pp. 41–44.
- [7] A.D. Calway, “Image representation based on the affine symmetry group,” in *IEEE ICIP.*, 1996, pp. 189–192.
- [8] A. Bhalerao and R. Wilson, “Warplet: An image-dependent wavelet representation,” in *IEEE ICIP.*, 2005, pp. 490–493.
- [9] C. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, 1995.

Chapter 2

Image Denoising

2.1 Introduction

Directional information is an important feature of images, and is exploited in many image processing applications. This chapter presents an empirical study of image denoising based on directional patterns. From the literature, the directional Wavelet transform, such as the Curvelet, has proven effective for denoising purposes. We introduce the Multiresolution Fourier Transform (MFT) facilitated with a multi-directional filter. From a statistical viewpoint, Independent Component Analysis (ICA) is drawing increasing attention. A combined solution is presented that benefits from the efficiency of the MFT and the data adaptiveness of ICA. Experimental results are very encouraging, even compared with other recent directional transforms such as the Curvelet.

Removal of noise from noisy images to obtain the unknown original image is often referred to as denoising. Gaussian additive white noise has a frequency spectrum that is continuous and uniform over a specified frequency band. It is spatially uncorrelated, and

the noise for each pixel is independent and identically distributed (iid).

$$\bar{f} = f + \xi \quad (2.1)$$

where \bar{f} is the noisy image, f is the original image and ξ is i.i.d. noise. Images are assumed to be linear-shift invariant, and linear methods such as the Wiener filter and the Kalman filter are often employed for denoising. Linear denoising methods are simple and inexpensive to implement, however they tend to blur the edge structure of the image, structure that is very important to the human visual system. The Markov random field, as well as various extensions, has been utilised to model the contextual information embedded in image formation. Partial differential equation (PDE)-based techniques (1) have also attracted much attention recently, in which image details are preserved by adding an edge detection term. The Single Value Decomposition method decomposes the column space of the observation matrix into a dominant and a subordinate part, revealing which of its subspaces can be attributed to the noise-free signal and which can be attributed to the noise (2). It is often assumed that these two subspaces are orthogonal to each other, which implies that signal and noise are independent. In a similar manner, a blind source separation or the Independent Component Analysis (ICA) decomposes signals assuming the following relation between components (3).

$$P(A \wedge B) = P(A) \cdot P(B) \quad (2.2)$$

where A and B are the independent components of signal. Given sufficiently large number of components, a few of the components can be pure noise components.

The Wavelet transform can decompose the original signal into a smooth part (lowpass) and a detailed part (highpass). For most signals, energy is mainly distributed in the smooth subband, and energy in the detail subband is clustered to a few large wavelet coefficients, corresponding to the edge structure of the original signal. Donoho and his colleagues (4)

pioneered a wavelet denoising scheme by using soft thresholding and hard thresholding. This approach, with the orthonormal wavelet, thresholds the wavelet transform coefficients within the detail subband. It is well known that Donoho's method offers the advantages of smoothness and adaptation. However, as Coifman and Donoho pointed out, this algorithm exhibits visual artefacts: Gibbs phenomena in the neighbourhood of discontinuities. However, the fundamental limitation of the orthonormal wavelet transform is the limited directional subband regardless of scale. In response, Starck and colleagues (5) proposed the Curvelet transform that extracts directional features in multi-scale using the Ridgelet. Later, other similar transforms followed such as the Contourlet which also consists of directional filter banks (6). These transforms are proven effective in denoising. Compared with these new techniques, a much older technique with inherent directional feature recognition can achieve similar performance with a directional filter, that is the Multiresolution Fourier Transform (MFT). We present the ICA-MFT combined algorithm as well as the MFT with a multi-directional filter.

The chapter is organised as follows. The next section starts with an introduction to ICA in the context of the denoising task. In section.2.3, denoising using the MFT with a Gaussian mask is explored and a combined approach is described in section.2.4. Section.2.5 introduces the MFT facilitated with a multi-directional filter based on Radon analysis. Section.3.4 reports an experimental comparison with other state-of-the-art techniques (5).

2.2 ICA Based Approach

Independent Component Analysis (ICA) has been frequently applied to computational neuroscience and the modelling of simple and complex cells in the human primary visual cortex (V1) (3; 7) which is responsible for directional feature identification. Recently, an ICA based denoising method has been developed by Hyvarinen and his colleagues (8). The basic motivation behind this method is that the ICA components of many signals are often very

sparse so one can remove noise in the ICA domain.

$$x = As \tag{2.3}$$

where x , A and s are the observed data, a linear mixing matrix and the source (latent) data respectively, which are independent and nongaussian. Unlike the wavelet-based denoising methods, an ICA based method uses a representation that is estimated solely by the statistical properties of the available data. The estimation of the ICA data model can be reduced to the search for uncorrelated directions in which the components are as nongaussian as possible and as a result the independent components have a sparse (supergaussian) distribution as possible. Hyvarinen developed a sparse code based noise shrinkage method similar to the wavelet shrinkage method (8).

$$x = As + v \tag{2.4}$$

where v is a Gaussian noise vector and x the noisy signal. An approximate version of s , \hat{s} can be obtained, applying shrinkage on $\hat{A}^{-1}x$. The components, $\hat{A}^{-1}x$ (neuron from a physiological viewpoint) with small activities are assumed as noise and shrunken, retaining only a few components with large activities (8). This, however differs in the following aspects. The shrinkage nonlinearities are estimated separately for each component, as opposed to a single fixed model in wavelet shrinkage. Also maximum-likelihood estimation is used in the nonlinearity estimation instead of minimax estimation. This method assumes training to estimate the orthogonal basis with noise-free data that has similar statistical properties. However, we attempted denoising without prior-training with empirically optimized settings.

2.3 MFT Based Approach

The ability to capture the directional patterns which exist at various locations, scale and orientation is a recent research trend in the image processing community. For example, the curvelet (5) represents a curve as a superposition of functions of various lengths and width controlling orientation across various scales. Various implementations have been proposed such as the curvelet, contourlet, brushlet, etc. The same ability can be found in the Multi-resolution Fourier Transform (MFT) (9). The MFT has been proposed as a combination of STFT and the wavelet. With the windowing function $g(t)$, the transform of a function $f \in L^2(R)$ at position u , frequency ξ and scale s is defined as below.

$$Mf(u, \xi, s) = \frac{1}{\sqrt{s}} \int_{-\infty}^{+\infty} f(t)g(s(t - u))e^{-i\xi t} dt \quad (2.5)$$

The Laplacian pyramid is used to decompose the image according to frequency which shows isotropic behavior. At each scale, the windowed Fourier transform is applied with the same window. The high frequency directional patterns can be observed in the Fourier local spectrum. This is where an elliptical shape of Gaussian filtering is suggested in (10) as follows.

$$G(x) = \frac{1}{2\pi} \left(\frac{-x^T C^{-1} x}{2} \right) \quad (2.6)$$

where C is a covariance matrix, which can be obtained from the inertia tensor of the spectrum. Experimental denoising results are presented in section.3.4.

2.4 ICA-MFT Combined Approach

The methods in the previous two sections approach the denoising problem in a different way. The ICA-based method uses purely statistical properties of the available data, adapting to the data. As a result, it requires larger computation even without a training process, as the specified number of bases increases. The second approach takes advantage of the Fourier

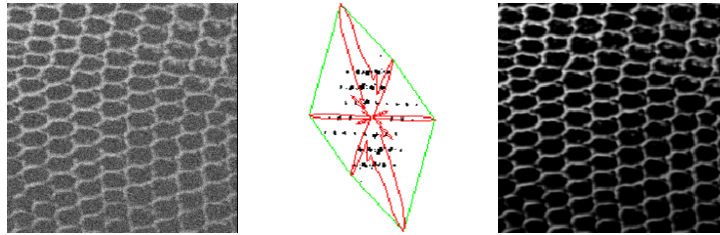


Figure 2.1: Radon-MFT Denoising (noisy image, a multi-directional filter on spectrum and denoised image)

transform of which the magnitude spectrum exhibits a directional energy pattern. The nonlinearity of the Gaussian function formed by the inertia tensor from the spectrum is used for soft-thresholding. Yet, it is difficult to represent a multi-directional pattern with a single Gaussian model. A Gaussian mixture model could be employed (11), but its estimation can increase the computational burden and also there is always the local minima problem. These two methods can be combined in a way that an algorithm can be computationally efficient and also adaptive to data, in two steps.

1. Perform ICA on each subband of the MFT with a limited number of components specified, for instance one third of the number of components needed than when ICA is performed alone. This ratio needs to be determined adaptively and accurately so that no component with a meaningful pattern is not discarded. The discarded components are assumed to represent pure noise. This reduces the computation significantly.
2. Apply a 2D Gaussian filter obtained from the Fourier spectrum of the bases found in step.1. As most of the components are localised in orientation as well as frequency, a much narrower Gaussian model that fits the data can be obtained from the spectrum.

2.5 MFT-Radon Approach

The last approach benefits most from the fact that the ICA decomposes a block of a specific frequency band comprising multi-directional features into a set of bases localized in orienta-

Table 2.1: SNR Results for different noise levels

	Noise(dB)	Curvelet	DDCT	TIWP	MFT	ICA	ICA-MFT	Radon-MFT
lena	5	17.55	17.44	18.91	18.97	18.74	19.50	19.43
	10	18.67	18.56	21.05	21.61	19.51	21.32	21.82
	15	19.40	19.37	23.49	24.42	21.48	22.65	24.55
	20	19.83	19.80	26.21	26.80	22.98	23.01	27.67
jaguar	5	12.19	12.18	15.41	15.45	12.24	12.66	13.36
	10	12.88	12.84	18.11	17.90	12.66	13.71	17.72
	15	13.24	13.26	20.94	19.89	13.28	15.23	21.04
	20	13.43	13.42	23.85	21.07	14.38	14.59	23.54

tion, so that the elliptical shape of the Gaussian filter obtained from the local spectrum fits the Fourier spectrum of the basis well. In this way, the inherent denoising capability of ICA discussed earlier combines with that of the MFT. Alternatively, however, we can replace ICA by providing a multi-directional filter using the Radon transform as shown in Fig.2.1. The Radon transform integrates a 2D function, f over the set of lines that are at a distance, s from the origin with the angle, θ between the line and the axis. The transform is related to the Fourier transform by the projection-slice theorem. A slice through the origin of the Fourier transform of the function, f , which is parallel to the projection line, is equal to the Fourier transform of the Radon projection of the function onto a line, $s = 0$. Therefore, the Radon projection on the local spectrum can form a closed shape, S consisting of a set of point as below

$$S = \{[r(\theta)\cos\theta, r(\theta)\sin\theta]^T\} \quad (2.7)$$

$$r(\theta) = c_i \cdot \Sigma\Sigma|F(x, y)|\delta(x\cos\theta + y\sin\theta) \quad (2.8)$$

where c_i indicates a normalization constant at MFT scale i , and F is the Fourier spectrum. The shape is used instead of the elliptical shape for the Gaussian filter for hard-thresholding, i.e. coefficients outside the shape are zeroed.



Figure 2.2: Comparative test : Lena

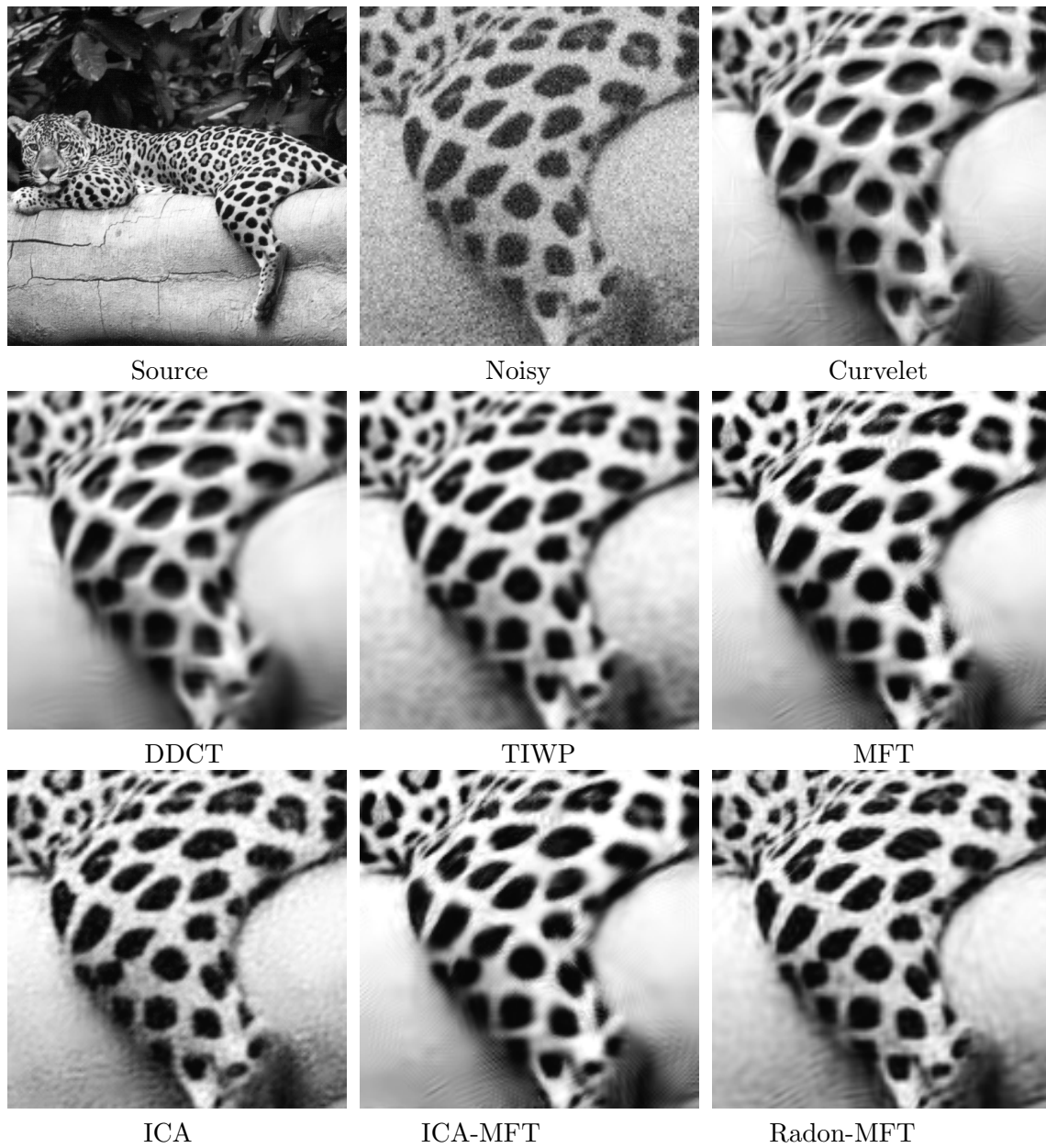


Figure 2.3: Comparative test : Jaguar

2.6 Experimental Results

The MFT was implemented on the Laplacian pyramid with 3 scale levels and a 50% overlapping \cos^2 window of 16×16 . For the Radon-MFT, the window size is increased to 32×32 for a better directional analysis. We tested ICA, ICA-MFT and Radon-MFT against other directional basis transforms, i.e. Curvelet (5), Directional Cosine Transform (DDCT) (10), MFT-Gaussian Filtering(MFT)(10) and Translation Invariant Wavelet Packet (TIWP). The test images are **Lena** and **Jaguar** as shown in Fig.2.2 and Fig.2.3, with comparative results on a noisy image of 15 dB. In Fig.2.2, **Lena** has a region of fur/feathers on her hat that creating multi-directional patterns, while the rest of the image is either homogeneous or directional. Clearly most of the methods with directional bases preserve the texture while TIWP produces quite a blurred image. The MFT, ICA-MFT and Radon-MFT show better results. **Jaguar** consists of a blob pattern on the jaguar skin. The Curvelet and DDCT suffers from the Gibb's phenomenon while both ICA based approaches show rather blurred images. Because of the various orientations in the blob jaguar texture, most ICA bases are used for orientation. The MFT and Radon-MFT generally show better results with much lower computational cost than the other methods. This is true for both test images. The Radon-MFT preserves the multi-directional pattern better than the MFT. The results of experiments with various noise levels are presented numerically (SNR) in Table.2.1 (whole image). In **Lena**, Radon-MFT shows good SNR result and preserves edge structures as shown in Fig.2.2 and Fig.2.3. In **Jaguar**, despite TIWP shows good SNR result, the edge structures are quite blurred.

2.7 Conclusion

There are many approaches to tackle image denoising. Wavelet shrinkage has proven effective in denoising various kinds of image. An extension of the Wavelet transform with directional subbands became popular as it can isolate directional singularity. ICA has also

been found useful in denoising. It has a different theoretical background, however, it is linked to the directional basis transforms as ICA shares the localization of orientation of bases. The main difference stands out in how the bases are obtained. The Wavelet is a self-similarity transform (like the Fourier transform) in which a set of source waveforms is predefined. On the other hand, ICA finds bases from the available data set using the independence relationship. We have reported a combined approach that is efficient and adaptive to the data, utilizing the fact that the ICA basis is localized in orientation. We have also developed another method replacing the ICA by multi-directional filtering in the MFT. In general it has shown better denoising performance compared with other directional basis transforms, with much lower computational requirement.

Bibliography

- [1] P. Perona and J. Malik, “Scale-space and edge detection using anisotropic diffusion,” *IEEE Trans. on PAMI.*, vol. 12, no. 7, pp. 629–639, 1990.
- [2] K. Konstantinides, B. Natarajan, and G.S. Yovanof, “Noise estimation and filtering using block-based singular value decomposition,” *IEEE Trans. IP.*, vol. 6, no. 3, pp. 479–483, 1997.
- [3] A. Hyvarinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, John Wiley, 2001.
- [4] D.L. Donoho, “Denoising by soft-thresholding,” *IEEE Trans. IT.*, vol. 41, pp. 613–627, 1995.
- [5] J.L Starck, E.J Candes, and D.L Donoho, “The curvelet transform for image denoising,” *IEEE Trans. on Image Processing*, vol. 11, pp. 670–684, 2000.
- [6] M. N. Do and M. Vetterli, “The contourlet transform: an efficient directional multiresolution image representation,” *IEEE Trans. on Image Process.*, vol. 14, no. 12, pp. 2091–2106, 2005.
- [7] B.A. Olshausen, “Learning sparse, overcomplete representations of time-varying natural images,” in *IEEE ICIP.*, 2003, pp. 41–44.
- [8] A. Hyvarinen, “Sparse code shrinkage: Denoising of nongaussian data by maximum likelihood estimation,” *Neural Computation*, vol. 11, no. 7, pp. 1739–1768, 1999.
- [9] R. Wilson, A.D Calway, and E.R.S Pearson, “A generalized wavelet transform for fourier analysis: themultiresolution fourier transform and its application to image and audiosignal analysis,” *IEEE Trans. on Image Processing*, vol. 38, pp. 674–690, Mar. 1992.
- [10] Y. Zhen, N. Rajpoot, and R. Wilson, “Directional wavelet with fourier-type bases for image processing,” in *Wavelet Anaysis and Applications*, 2005.
- [11] H. Park, A. Bhalerao, G.R. Martin, and A.C. Yu, “An affine symmetric approach to natural image compression,” in *Mobimedia*, 2006.

Chapter 3

Image Segmentation

3.1 Introduction

Many natural textures comprise structural patterns and show strong self-similarity. We use affine symmetry to address the self-similarity relationship; that is a patch of texture can represent other similar patches after affine transformation. Based on this model, we demonstrate efficient segmentation of structural textures. The algorithm employs the Multi-resolution Fourier Transform and uses Radon analysis to obtain a directional shape from the local spectrum. This process offers reasonable noise robustness for directional feature detection. From the obtained shape, a feature is extracted using the affine-invariant Fourier descriptor. Experimental evaluation on structural textures shows encouraging results and application on natural images demonstrates identification of texture objects.

Image segmentation is a well-known problem in the image processing and computer vision community. Its application covers a wide range of areas such as tracking, surveillance, medical imaging, robot navigation and many more. Texture plays an important role in accurate image segmentation. Segmentation itself is a mathematically well-defined task that splits and joins groups of pixels depending on whether they share the same feature. Texture

segmentation has been studied for decades (1; 2; 3) but some problems remain unresolved due to the inherent difficulties. Texture is an ambiguous term. There is no clear definition but one might suggest primitive patterns that are repeated in a periodic manner. There are two prominent approaches to define texture. One approach defines a primitive individual element, *texton* that comprises texture (4). The other approach (2; 3) derives primitive statistical feature vectors from a group of pixels where the visual element is difficult to observe, for example the texture of grass. In this paper, we consider the special class of texture that exhibits a structural pattern such as reptile skin, brick wall and Jaguar blobs. Our approach is similar to that of the *texton*. We assume that texture is a region that is self-correlated by affine symmetry and a prototypical block can represent the remainder of region. Previously, Wilson and Li (1) showed texture segmentation based on the same principle and demonstrated promising results. They used the residue after affine transformation between blocks as a distance metric in the clustering. This method requires considerable computation as the affine transform of all possible pairs of blocks is calculated. Later, Bhalerao and Wilson reduced the computation by using the Fourier spectrum as a feature vector and having multiple deformed centroids of the same prototypical block by scale and orientation (5). However, both approaches require manual selection of the initial prototypical blocks. Park et al (6) showed unsupervised block classification based on the number of directional features using independent component analysis (ICA). We present a texture segmentation algorithm that is affine invariant, requires comparatively little computation and is unsupervised. This chapter is organized as follows. Section.3.2 covers affine invariant feature extraction. Section.3.3 illustrates the overall process. Section.refexperiments reports the experimental results. Lastly, conclusions are drawn.

3.2 Affine Invariant Features

We present an efficient (short) and affine invariant feature based on directional information. We employ the Multi-resolution Fourier Transform with $16 \times 16 \cos^2$ window and use the high frequency band that exhibits pure patterns. The directional information is extracted from the local spectrum using Radon analysis. The extraction proceeds as follows: (1) Fourier transform (2) Directional shape Radon analysis (3) Affine invariant Fourier shape description. The convex hull area ratio, which will be explained later, is not taken as a feature, however it can indicate the presence of strong directional patterns and can be used for selecting a representative block in each group.

3.2.1 Fourier Transform Spectrum

A general affine transformation AT in \mathbb{R}^2 space is defined as

$$AT(x) = Ax + t \quad (3.1)$$

The Fourier transform is well utilized in (1; 5) to separate the affine transform, AT , into a linear part, A , which affects only the magnitude spectrum and a translational part, t that is exhibited as a phase gradient. One can observe a linear affine deformation of the magnitude spectrum such as rotation, scaling and shearing between blocks, regardless of the translational difference. We extract directional information from the local magnitude spectrum, which is then described by an affine invariant descriptor.

3.2.2 Directional Shape Radon Analysis

The radon transform, RT , integrates a 2D function, f over the set of lines that are a distance, s from the origin with the angle, θ between the line and the axis.

$$RT(\theta, x) = \Sigma \Sigma f(x, y) \delta(x \cos \theta + y \sin \theta - s) \quad (3.2)$$

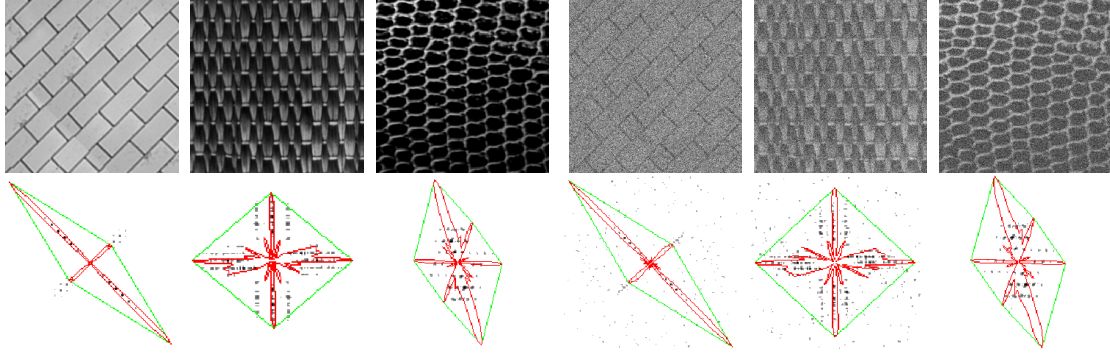


Figure 3.1: Structural texture samples: brick, weave, reptile and the same sample with white Gaussian noise (5dB) (top) and Fourier magnitude spectrum of the texture and extracted shape (bottom)

The transform is related to the Fourier transform by the projection-slice theorem. A slice through the origin of the Fourier transform of the function, f , which is parallel to the projection line, is equal to the Fourier transform of the Radon projection of the function onto a line. We use the Radon transform as a means of analysing the spectrum. As the projection-slice shows, the Fourier spectrum can uncover the directional pattern of the source. A projection of slices at multiple angles evenly distributed around a resolution, $[0\ 2\pi]$ can reveal a directional pattern of the function by forming a closed shape, S consisting of a set of points as below:

$$S = \{[r(\theta)\cos\theta, r(\theta)\sin\theta]^T\} \quad (3.3)$$

$$r(\theta) = \frac{\Sigma\Sigma|F(x, y)|\delta(x\cos\theta + y\sin\theta - s)}{e^{\sigma^2}} \quad (3.4)$$

where c and σ^2 indicate a normalization constant and variance of the Fourier slice, respectively. Fig.3.1 shows the sample texture in the top row and the extracted directional shapes. The right three texture samples are identical to those on the left except for the addition of white Gaussian noise (5dB). As shown, the extracted shapes (red) are consistent regardless of the presence of noise.

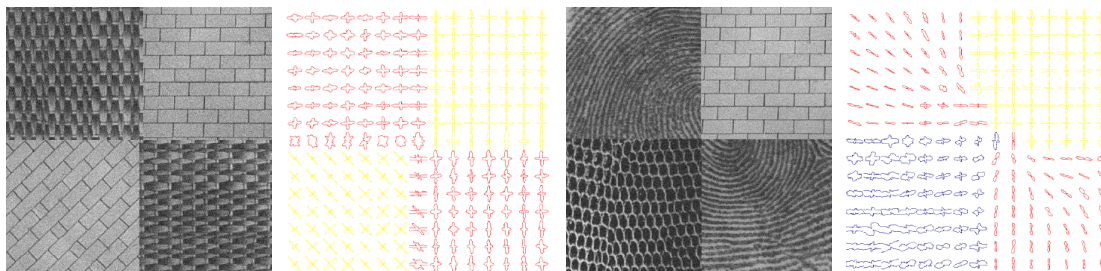


Figure 3.2: Segmentation test of texture composites: texture composite and classification result in pair, both texture samples are added with white Gaussian noise (15dB)

3.2.3 Affine Invariant Shape Descriptor

Given a shape, S , its constituent boundary pixels, C , $[x(t), y(t)]^T$ are traversed to yield a parametric equation based on the affine length of a closed curve, as below. This is linear under affine transformation and also yields the same parameters, independent of the initial representation.

$$\int \sqrt[3]{\dot{x}(t)\ddot{y}(t) - \dot{y}(t)\ddot{x}(t)} dt \quad (3.5)$$

where the number of dots indicate the order of the derivatives. Having encoded the boundary as a function of the parameter, taking the Fourier transform of the boundary equation results in $[U, V]^T$, where U and V are Fourier coefficients referring to the x and y coordinates respectively. Since the Fourier transform is a linear operator, the equation below holds

$$[U_k, V_k]^T = A[U_k^0, V_k^0]^T \quad (3.6)$$

where $[U_0, V_0]^T$ denotes the same coefficients from the affine transform of the reference block. By including another coefficient and extending eq. (6) to a 2×2 matrix, obtaining their determinants reveals a linear factor. A simple division of both sides by either side produces an absolute affine invariant feature. For more details, readers are referred to (7).

3.2.4 Convex Hull - Area Ratio

The convex hull area ratio is another affine invariant metric. Although this metric is not used as a feature in our algorithm, it can reveal whether strong direction patterns are present in a sample block. This follows the same idea as in (6), where a simple isometric measure based on PCA was used to determine the most representative block in a class under two observations. One is choosing block in a same class will not make significant improvement to approximate other blocks. The other is that only one directional singularity is likely to exist as the window size gets smaller (16×16). Nevertheless, the metric does not work well in the case of multiple directional features. The convex hull is obtained efficiently by a Grahams scan. Given a set of points, s , the convex hull area ratio, R is given as

$$R(s) = \frac{Area(s)}{Area(ConvexHull(s))} \quad Area([x, y]^T) = \frac{1}{2} \sum_{i=0}^{N-1} x_i y_{i+1} - x_{i+1} y_i \quad (3.7)$$

where N denotes the number of points. The convex hull of the shape is shown in (green) in Fig.3.1.

3.3 Affine Invariant Classification

This section shows the overall workflow of our segmentation algorithm. Regarding the window size for texture analysis, there is a problem known as the class-boundary uncertainty i.e. if we confine the analysis to a small window, we get a better resolution of segmentation but we lose confidence of texture characteristics within the window. On the other hand, a bigger window allows a better analysis of texture but results in a coarse resolution of segmentation. An adequate solution is a multi-resolution approach. Texture information at the top level passes on to the lower level and is combined recursively as a top-down approach. We use the Multi-resolution Fourier Transform (MFT) as given below(8). For a

given scale(s), frequency(ξ), position(u) and image $f(x)$, the 2D MFT is defined as,

$$Mf(u, \xi, s) = \frac{1}{\sqrt{s}} \int_{-\infty}^{+\infty} f(t)w(s(t-u))e^{-i\xi t} dt \quad (3.8)$$

where w denotes an appropriate window function as given below.

$$w(x) = \cos^2[\pi p/2B]\cos^2[\pi q/2B] \quad (3.9)$$

where $x = [p, q]^T$ and B is block size. Having applied the MFT on a source image, the affine invariant features are extracted from the Fourier magnitude spectrum at each scale using directional shape radon analysis followed by an affine invariant shape descriptor. The resultant features, v at the bottom scale are joined together with the quadtree parent as follows:

$$Feature(i, j) = \{p^0(V), p^1(V), \dots, p^k(V)\} \quad (3.10)$$

$$p^k(V_{i,j}) = V_{\lfloor i/2^k \rfloor, \lfloor j/2^k \rfloor, k} \quad V_{i,j,k} = \omega_k \times \{v_{i,j,k}\} \quad (3.11)$$

where ω^i and k refer to a weight and scale respectively. The combined features are fed into a K-means clustering algorithm. Prior knowledge of the number of classes is required due to the nature of the algorithm. The K-means clustering is chosen only for simplicity as our main focus is to determine an effective affine invariant feature. A random field approach may be of interest if convergence of the number of classes is desired (1; 9).

3.4 Experiments

We intended the affine-invariant features to be robust to noise. The algorithm was tested using simple composites of sample structural textures to which Gaussian noise was added (15dB), as shown in Fig.3.2. Firstly, it is observed that the directional shapes obtained from the Fourier transform were too jagged due to the scattered high-frequency coefficients,

which in turn disrupts the affine invariant description and results in a poor clustering. Consequently, the scattered high-frequency coefficients are removed using minimax thresholding before the Radon projection and the Gaussian smoothing filter is applied to the shape boundary. This, in fact, makes the shape-extraction robust to noise. The classification result shows the extracted shapes that are colored according to classes in Fig.3.2. Secondly, many shapes from the bottom level (16×16) of the MFT were elliptical, which makes affine invariant shape description useless. Starting with a window of 32×32 at the bottom level still produces an acceptable result. However, we have found that using the size of the shape at the bottom level feature produces a better result. This is because the shape size increases with the strength and directionality of the feature and decreases as the directional pattern becomes less significant. The clustering algorithm orders them by shape area. It is probable that different textures with the same number of directional features fall into the same class, but it is the most information we can gather at the bottom level. The structural information from a bigger window is passed on from the parent block and combined together as discussed Section.3.3. Having successfully tested the algorithm on the sample test image set, the natural images, *jaguar* and *zebra* were used. Each of the natural test images has a different characteristic texture. Fig.3.3 shows the original image, block classification result (3 classes), segmented texture and reconstruction. Jaguar blob texture is segmented well, but there are some holes in zebra stripes segmentation that are resulted from the smaller window size than the bold strips of zebra. To see if the block classification result is indeed affine symmetric, the image is reconstructed by selecting a prototypical block in each class and replacing other blocks by the affine transformed prototype. The affine model is estimated using the two-component method (10; 11; 12; 6) and the prototype blocks were chosen by eq.(3.7). A non-affine invariant classification would have resulted in an odd image, for instance, the log featuring blobs of the jaguar. The reconstruction quality varies around 25dB in PSNR and is visually acceptable. This work not only presents an interesting approach to the segmentation task but also offers a feasi-



Figure 3.3: Block classification test on natural images: jaguar (top) and zebra (bottom), original image, classification result, segmentation and reconstruction (by column)

ble solution for efficient implementation. We intend to employ this algorithm in our affine image coding system described in (6).

3.5 Conclusion

Affine invariance has been applied to segment self-similar texture previously by many researchers. However, the complexity of the algorithms has prohibited their practical implementation. Our motivation has been to develop a computationally efficient image texture classification algorithm while maintaining the texture discrimination power of previous approaches (1; 5). We have demonstrated a simple and efficient approach utilizing affine invariant shape description. Experimental evaluation indicates acceptable segmentation results for structural texture such as brick wall, jaguar and reptile skin, and also the algorithms robustness to noise. Further study using a random field may improve results. The presented work is of interest where efficient texture segmentation is demanded.

Bibliography

- [1] R. Wilson and C.T. Li, “A class of discrete multiresolution random fields and its application to image segmentation,” *IEEE Trans. PAMI*, vol. 25, no. 1, pp. 42–56, 2002.
- [2] R.M. Haralick, K. Shanmugam, and I. Dinstein, “Texture features for image classification,” *IEEE Trans. SMC*, vol. 3, no. 6, pp. 610–621, November 1973.
- [3] R.M. Haralick, “Statistical and structural approaches to texture,” *IEEE Proc.*, vol. 67, no. 5, pp. 786–804, May 1979.
- [4] B. Julesz, “Texture gradients: The texton theory revisited,” *Spatial Vision*, vol. 1, no. 1, pp. 19–30, 1985.
- [5] A. Bhalerao and R. Wilson, “Affine invariant image segmentation,” in *BMVC.*, 2004.
- [6] H. Park, A. Bhalerao, G.R. Martin, and A.C. Yu, “An affine symmetric approach to natural image compression,” in *Mobimedia*, 2006.
- [7] K. Arbter, W.E. Snyder, H. Burkhardt, and G. Hirzinger, “Application of affine-invariant fourier descriptors to recognition of 3-d objects,” *IEEE Trans. PAMI.*, vol. 12, no. 7, pp. 640–647, 1990.
- [8] R. Wilson, A.D Calway, and E.R.S Pearson, “A generalized wavelet transform for fourier analysis: themultiresolution fourier transform and its application to image and audiosignal analysis,” *IEEE Trans. on Image Processing*, vol. 38, pp. 674–690, Mar. 1992.
- [9] C.T. Li, “Multiresolution image segmentation integrating gibbs sampler ang region mergion algorithm,” *Signal Processing*, vol. 83, pp. 67–78, 2003.
- [10] A.D. Calway, “Image representation based on the affine symmetry group,” in *IEEE ICIP.*, 1996, pp. 189–192.
- [11] A. Bhalerao and R. Wilson, “Warplet: An image-dependent wavelet representation,” in *IEEE ICIP.*, 2005, pp. 490–493.
- [12] T.I Hsu and R. Wilson, “A two-component model of texture for analysis and synthesis,” *IEEE Trans. on Image Processing*, vol. 7, no. 10, pp. 1466–1476, October 1998.

Chapter 4

Future work

4.1 Image Coding

The work presented has shown a direction in which affine symmetric image coding can compete with a current image coding system, the Wavelet transform image coding, at very competitive bitrates. At the current stage, however, the coder has a high computational complexity due to the block clustering process involving ICA, and also the clustering is not accurate enough. The clustering is an essential step in our algorithm that classifies blocks by affine-symmetry. An efficient approach is required that can replace ICA. We plan to deploy our segmentation algorithm for that purpose. Secondly, the bit-allocation problem should be addressed so that natural quality degradation is allowed as the bitrate decreases.

4.2 Image Denoising

The work presented has shown a combined approach of a self-similarity transform and blind source separation (or computational efficiency versus adaptiveness). In this approach, ICA is performed with a limited number of components, resulting in significant computation reduction, and then an oriented Gaussian filter is applied on the bases. Also, we suggested an efficient approach utilising a multi-directional filter. Denoising is not only an integrated

functionality in our image coder but also the understanding behind denoising aids solving the bit-allocation problem in image coding. This work can be useful to optimise rate-distortion of the image coder.

4.3 Image Segmentation

We have demonstrated a simple and efficient approach utilizing affine invariant shape description. Experimental evaluation indicates acceptable segmentation results for structural texture such as brick wall, jaguar and reptile skin, and also the algorithm's robustness to noise. Application on certain natural images shows identification of texture object. This is very encouraging and potentially the technique can be used in image retrieval and indexing applications. The algorithm is also a good candidate to be employed in image coder since it is simple and computationally inexpensive. However, work still remains to make the algorithm more robust to various kinds of images.

4.4 Video: Motion Estimation

Motion estimation is a natural application from affine-symmetry theory. Current video motion estimation in standard video codecs assumes rigid movement which fails to capture real motion such as rotation. Mesh based motion estimation is a more advanced technique, however, sharing each grid point with its neighbouring nodes constrains a full realisation of the affine transform. Due to grid points connected to multiple adjacent triangles, in effect it is impossible to describe even a simple rotation perfectly(1). Furthermore, the estimation process is expensive. We plan to use a similar approach to that used in section 3.1 along time axis. The resulting parameters can be excessive, thus an appropriate parameter coding technique should also be developed. Distributed video coding is worth investigating for this purpose (2; 3).

4.5 Video: Texture Synthesis

Texture synthesis is one possible applications of affine symmetric segmentation as the segmentation result allows one to determine which block can be simply substituted by another similar block with slight deformation. Having performed image segmentation and sharing its results with the decoder, one might send a short indication to save bits to encode a part of a certain texture by substituting it with one already received. This approach is already being tested by many researchers and regarded as a feasible solution to further improve the compression performance of image coding as well as of video coding.

Bibliography

- [1] Heechan Park, A. C. Yu, and G. Martin, “Progressive mesh-based motion estimation using partial refinement,” in *LNCS*, Apr. 2006, pp. 98–101.
- [2] R. Puri and K. Ramchandran, “Prism: An uplink-friendly multimedia coding paradigm,” in *IEEE ICASSP.*, Apr. 2003, pp. 856–859.
- [3] R. Puri and K. Ramchandran, “Prism: A new robust video coding architecture based on distributed compression principles,” in *Communication, Control and Computing*, Allerton, IL, Oct. 2002.