

Anisotropic Channel Filtering

Michael Felsberg* and Gösta Granlund

Computer Vision Laboratory,
Department of Electrical Engineering,
Linköping University, SE-58183 Linköping, Sweden,
mfe@isy.liu.se, gosta@isy.liu.se,
<http://www.isy.liu.se/cvl>

Abstract. Channel smoothing is an alternative to diffusion filtering for robust estimation of image features. Its main advantages are speed, stability with respect to parameter changes, and a simple implementation. However, channel smoothing becomes unstable in certain situations, typically for elongated, periodic patterns like for instance fingerprints. As for the diffusion filtering an anisotropic extension is required in these cases. In this paper we introduce a new method for anisotropic channel smoothing which is comparable to coherence enhancing diffusion, but faster and easier to implement. Anisotropic channel smoothing implements an orientation adaptive non-linear filtering scheme as a special case of adaptive channel filtering. The smoothing algorithm is applied to several fingerprint images and the results are compared to those of coherence enhancing diffusion.

1 Introduction

In this paper we introduce a method for anisotropic, nonlinear smoothing of images. The results of our method are similar to those of coherence enhancing anisotropic¹ diffusion, see e.g. [1]. The new approach is based on the channel representation [2–4]. As shown in [5, 6], linear smoothing of channels is equivalent to a robust estimator and non-linear diffusion. As we will show, an extension to a sophisticated anisotropic smoothing technique is possible.

1.1 Channel Representation

The idea of the channel representation is the following. Instead of storing the value (or intensity) of a feature (e.g. grey level, orientation), the feature range is sampled with local (compact) smooth functions, see Fig. 1. The sampled values are combined to form the channel vector $\mathbf{c} = (c_{\min} \dots c_{\max})^T$, representing the feature.

* This work has been supported by DFG Grant FE 583/1-2

¹ Throughout this paper 'anisotropic' means anisotropic in the image plane, i.e., anisotropic diffusion uses a tensor valued diffusivity.

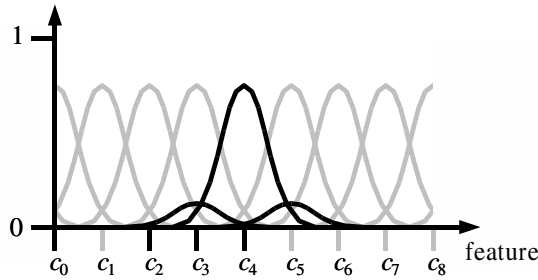


Fig. 1. Channel representation of a feature value. All channels except for c_3 , c_4 , and c_5 are zero. If the channel centers have unit distance, the encoded value is 4.

The components of this vector (the channel values) are monopolar (non-negative). In most frameworks, the channel values are chosen to be in the interval $[0, 1]$ and the channel vector is normalized, i.e., it can be considered as a sampled probability density function of the feature. Since the samples are obtained from local smooth functions which are monopolar and add up to one, the channel representation belongs to the kernel methods [7]. In general, the channel vector is sparse and only the non-zero elements need to be stored and processed. The non-zero channels are called active channels.

What are the benefits of the channel representation? First, it allows to store and separate multiple values. Hence, metamery [2, 8] is avoided to a certain degree depending on the basis functions and the sampling rate. Second, it is possible to perform non-linear operations on the original data by means of linear operations on the channel representation. A nice example for this linearization is the channel smoothing [9]. A third advantage is that the topology of the feature space need not be linear; for instance it can be periodic [10]. The channel encoding and decoding is hardly affected, whereas classical methods become much more complicated (e.g. orientation diffusion [11]).

In order to take advantage of the mentioned properties, it is necessary to have an appropriate *local* decoding available. In the present paper, we use quadratic B-splines [12] to compute the channel values. By a simple proof one can show that the decoding is given by a linear combination of the channel values [5]. Restricting the linear combination to a small interval of channel indices (the reconstruction window) yields a local reconstruction.

1.2 Channel Smoothing and Robust Estimators

When dealing with grey levels represented by channels, one might ask where the metamery might occur. One example where multiple valued functions occur is the local estimation of the mean value for non-stationary processes. Consider for example an edge in a noisy image, Fig. 2, upper left.

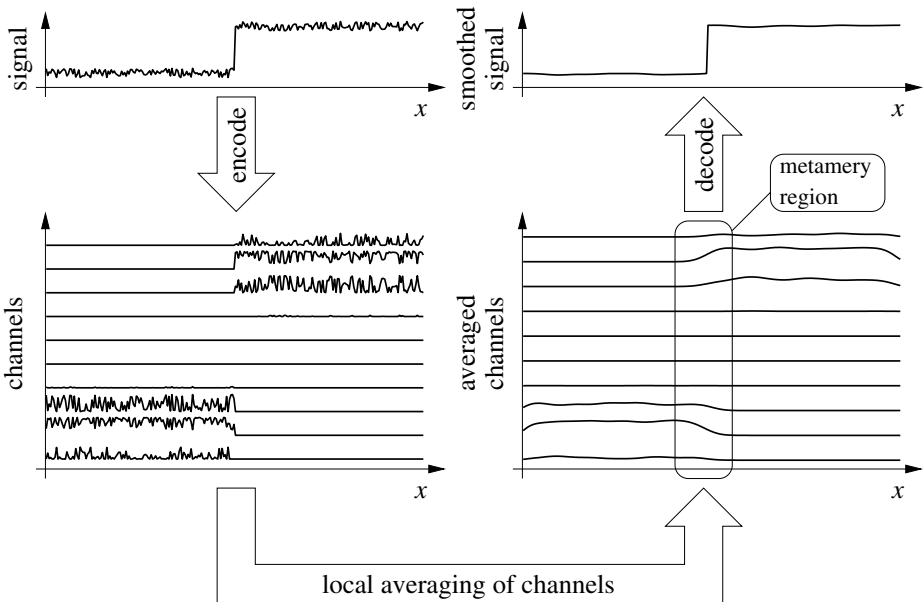


Fig. 2. Channel smoothing of a noisy edge (slice cut orthogonal to edge). Left: signal and its channel representation; right: averaged channels and reconstructed signal.

The aim of a sophisticated smoothing method should be to remove the noise, but not to blur the edge. Linear smoothing fails, since it implicitly considers the signal in the whole filter support to be stationary and therefore treats both sides of the edge as realizations of the same stochastic process. This is different for linear smoothing of channels: By an appropriate choice of the channel width and the sampling rate, one can assure that different grey levels are represented by disjunct channels. Linearly averaging of the channels therefore keeps the two sides of the edge separated, see Fig. 2, right.

By averaging the channels, the active channels are propagated across the edge (see Fig. 2, lower right). Hence, close to the edge and on the edge itself, the channels represent two different values at a time (metamery). Each of both is obtained by applying the local decoding, resulting in the two values and their corresponding likelihoods. These likelihoods are obtained by summing up the channel values in the respective reconstruction window. At every point the reconstruction with the highest likelihood is chosen [13]. The result is a signal which is smoothed within stationary regions only, Fig. 2, upper right.

The output of channel smoothing resembles the results from non-linear diffusion. Under certain assumptions, the latter is equivalent to a robust estimator [14] and in [5] it has been shown that channel smoothing is indeed a good approximation of a robust estimator. Therefore, channel smoothing is indeed an alternative to non-linear diffusion for noise removal in images and other feature

maps. The advantages of channel smoothing compared to diffusion is the lower computational complexity, a simpler implementation, and its applicability to non-linear spaces (e.g. orientation space [10]).

2 A Method for Anisotropic Channel Smoothing

Looking at the channel smoothing as such, it is not obvious, how to turn it into a real anisotropic smoothing. Applied to simple edges, the channel smoothing already shows an anisotropic behavior which is comparable to edge enhancing diffusion. Close to an edge, the filter support of a comparable linear filter is a semi-disc lying completely on one side of the edge. Hence, the averaging makes use of a support which can be compared to the one effectively used for edge enhancing diffusion. However, if we consider periodic, oriented patterns, which occur for instance in images of finger prints, the ordinary channel smoothing fails. The reconstructed image suffers from 'flip-over' effects, i.e., contours become fragmented or displaced.

2.1 Incorporating Orientation Information

Anisotropic smoothing requires orientation dependent processing [15]. Hence, the channel smoothing must be extended using orientation information. A quite elegant way to introduce orientation information is in the channel encoding step. The method works as follows.

1. For each position (x, y) the grey level is encoded in $\mathbf{c}_g(x, y)$.
2. A local orientation map is created, using the orientation of the gradient (modulo π).
3. For each position (x, y) the orientation is encoded in $\mathbf{c}_o(x, y)$.
4. The orientation channel representation is smoothed, in order to reduce the noise in the orientation information.
5. For each position (x, y) the channel matrix $\mathbf{c} = \mathbf{c}_g \mathbf{c}_o^T$ is computed.

The result is a channel representation where the orientation information is represented by the row index. The column index represents the grey levels, see Fig. 3. This extended channel representation can be decoded in three steps using the standard methods for linear and periodic decoding:

1. For each column in the matrix, decode the orientation.
2. The *likelihoods* of the decoded orientations form a new channel vector.
3. Apply the ordinary linear decoding to the new channel vector.

The sketched method works correctly, since for oriented structures the channel matrix has a clear 2D maximum. Projecting onto the grey level axis (which is basically what we are doing in step 1) yields a 1D channel vector with a 1D maximum at the appropriate grey level. For unoriented structures, we more or less randomly pick up grey level channels for arbitrary orientations. This is still correct, since all rows of the matrix are nearly identical in that case.

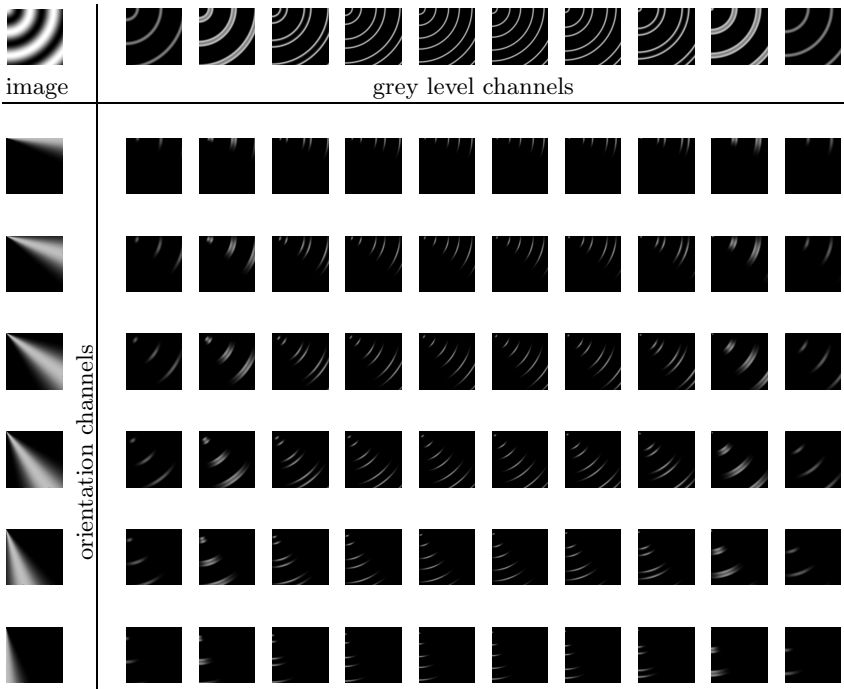


Fig. 3. Channel matrix using ten grey level channels and six orientation channels.

2.2 Anisotropic Smoothing

The channel matrix encodes explicitly which grey level is present at which orientation. Hence, it is straightforward to design a method for anisotropic smoothing. For ordinary channel smoothing, a linear smoothing operation is introduced between the encoding and the decoding. Each channel is smoothed with the *same* linear kernel. If the final result is supposed to be adaptive to the orientation, we simply use *differently oriented* anisotropic smoothing kernels, where we choose the orientation of the smoothing kernel according to the row index of the channel matrix.²

The anisotropic smoothing kernels which are applied in our method are polar separable filters. Their radial component is given by a Gaussian kernel and their angular component is given by the same B-splines which were used to encode the orientation before. Note that the sum over all orientations yields an isotropic Gaussian kernel.

² In this paper we ignore the possibility of choosing different kernels for different grey levels which would result in an intensity adaptive smoothing method [6]. Such a method would be appropriate if the assumed noise model is not additive.

For oriented structures, it is obvious why the previously described approach should work. The structure is smoothed perpendicular to the orientation vector³, but is hardly affected parallel to it. This should give very nice results for noisy, nearly periodic patterns with varying orientations, e.g., for fingerprints.

For unoriented structures, the anisotropic channel smoothing behaves similar to an isotropic method, which can be shown by a probabilistic consideration. In unoriented regions the orientation vector is equally distributed, i.e., in the reconstruction we randomly pick different rows of the channel matrix and therefore differently oriented smoothing kernels. Since the kernels sum to one, this corresponds to isotropic smoothing.

3 Experiments

In our experiments, we compared the presented method to the approach of coherence enhancing diffusion [16]. In the comparison we focused on two aspects: computational speed and the quality of the results. Both implementations are realized in Matlab, the coherence enhancing diffusion implementation was taken from [17]. We applied both methods to images of a fingerprint⁴, see Fig. 4.

The parameters for the anisotropic channel smoothing used for the results in Fig. 4 are: eight orientation channels, four grey level channels, isotropic smoothing kernel for orientation: $\sigma_i = 3$, and anisotropic smoothing kernel: $\sigma_a = 1.2$. The parameters for the coherence enhancing diffusion are taken from [17] (diffusion speed exponent: 0.001, scale of Gaussian derivative: 0.7, scale of structure tensor: 5, time-step: 0.2, iterations: 10, no display of results).

The anisotropic channel smoothing is three times faster than the coherence enhancing diffusion. Both methods are scarcely optimized, i.e., the speed-up factor might be slightly different for other implementations. The quality of the results is comparable, depending on the selection of the parameters. The actual values of the variances σ_i and σ_a are of little importance, i.e., the method is very stable with respect to changes in these parameters.

The parameters were chosen for the 300dpi images. Accordingly, the 600dpi images were not sufficiently smoothed and the 150dpi images are over-smoothed. Whereas for the insufficiently smoothed images the anisotropic channel smoothing and the coherence enhancing diffusion give quite similar results, the diffusion suffers more from aliasing in case of the over-smoothed case.

4 Conclusion

We have presented an anisotropic extension of channel smoothing which is comparable to coherence enhancing diffusion. The orientation information is explicitly considered by forming the outer product of grey level channel vector and

³ The orientation vector is for instance given by the normalized double angle gradient.

⁴ The images were acquired by repeatedly pressing the finger on an ink-pad and on a sheet of paper afterwards. The sheet of paper was then scanned with 150dpi, 300dpi, and 600dpi.

orientation channel vector. The smoothing kernels are adapted to the orientation information, such that the reconstructed signal is a robust estimate of the original signal with an orientation bias. Anisotropic channel smoothing is faster than coherence enhancing diffusion, but gives very similar results. It is very stable with respect to parameter changes and is easy to implement. Furthermore, it can easily be adapted to other features than orientation.

References

1. Weickert, J.: *Anisotropic Diffusion in Image Processing*. Teubner (1998)
2. Snippe, H.P., Koenderink, J.J.: Discrimination thresholds for channel-coded systems. *Biological Cybernetics* **66** (1992) 543–551
3. Nordberg, K., Granlund, G.H., Knutsson, H.: Representation and Learning of Invariance. In: *Proceedings of IEEE International Conference on Image Processing*. (1994)
4. Granlund, G.H.: An associative perception-action structure using a localized space variant information representation. In: *Proc. Int. Workshop on Algebraic Frames for the Perception-Action Cycle*, Springer, Heidelberg (2000)
5. Felsberg, M., Scharr, H., Forssén, P.E.: The B-spline channel representation: Channel algebra and channel based diffusion filtering. Technical Report LiTH-ISY-R-2461, Dept. EE, Linköping University (2002)
6. Scharr, H., Felsberg, M., Forssén, P.E.: Noise adaptive channel smoothing of low-dose images. In: *CVPR Workshop: Computer Vision for the Nano-Scale*. (2003) accepted.
7. Bishop, C.M.: *Neural Networks for Pattern Recognition*. Oxford University Press, New York (1995)
8. Koenderink, J.J.: What is a "feature"? *Journal of Intelligent Systems* **3** (1993) 49–82
9. Forssén, P.E., Granlund, G.H., Wiklund, J.: Channel representation of colour images. Technical Report LiTH-ISY-R-2418, Dept. EE, Linköping University (2002)
10. Johansson, B.: Representing multiple orientations in 2D with orientation channel histograms. Technical Report LiTH-ISY-R-2475, Dept. EE, Linköping University (2002)
11. Perona, P.: Orientation diffusions. *IEEE Transactions on Image Processing* **7** (1998) 457–467
12. Unser, M.: Splines – a perfect fit for signal and image processing. *IEEE Signal Processing Magazine* **16** (1999) 22–38
13. Forssén, P.E.: Sparse representations for medium level vision. Lic. Thesis LiU-Tek-Lic-2001:06, Dept. EE, Linköping University (2001)
14. Black, M.J., Sapiro, G., Marimont, D.H., Heeger, D.: Robust anisotropic diffusion. *IEEE Transactions on Image Processing* **7** (1998) 421–432
15. Knutsson, H., Wilson, R., Granlund, G.H.: Anisotropic non-stationary image estimation and its applications: Part I – restoration of noisy images. *IEEE Trans. on Communications* **COM-31** (1983) 388–397
16. Weickert, J.: Multiscale texture enhancement. In Hlaváč, V., Šára, R., eds.: *Computer Analysis of Images and Patterns*. Volume 970 of LNCS., Springer, Berlin (1995) 230–237
17. v. d. Boomgaard, R.: Nonlinear diffusion in computer vision. <http://carol.wins.uva.nl/~rein/nldiffusionweb/material.html> (2001) (Accessed 13 Jan 2003).



Fig. 4. Fingerprint experiment. From left to right: original images, results from diffusion, and results from channel smoothing. From top to bottom: fingerprint 1 (FP1) at 300dpi, zoomed detail of FP1, FP2 at 300dpi, FP1 at 600dpi, and FP1 at 150dpi.