A TV Flow Based Local Scale Measure for Texture Discrimination*

Thomas Brox and Joachim Weickert

Mathematical Image Analysis Group, Faculty of Mathematics and Computer Science,
Saarland University, Building 27, 66041 Saarbrücken, Germany
{brox,weickert}@mia.uni-saarland.de
www.mia.uni-saarland.de

Abstract. We introduce a technique for measuring local scale, based on a special property of the so-called total variational (TV) flow. For TV flow, pixels change their value with a speed that is inversely proportional to the size of the region they belong to. Exploiting this property directly leads to a region based measure for scale that is well-suited for texture discrimination. Together with the image intensity and texture features computed from the second moment matrix, which measures the orientation of a texture, a sparse feature space of dimension 5 is obtained that covers the most important descriptors of a texture: magnitude, orientation, and scale. A demonstration of the performance of these features is given in the scope of texture segmentation.

1 Introduction

Scale is an important aspect in computer vision, as features and objects are only observable on a certain range of scale. This is a consensus in the computer vision community for several years and has led to techniques that take into account multiple scales to reach their goal (scale spaces and multi-resolution approaches), or that try to automatically choose a good scale for their operators (scale selection).

As soon as *local* scale selection is considered, a measure for the local scale at each position in the image becomes necessary. The simplest idea is to consider the variance in a fixed local window to measure the scale. However, this has several drawbacks: large scales with high gradients result in the same value as small scales with low gradients. Moreover, the use of non-adaptive local windows always blurs the data. The latter drawback also appears with the general idea of local Lyaponov functionals [19], which includes the case of local variance. Other works on scale selection can be found in [10,11,12,5, 9]. All these methods have in common that they are gradient based, i.e. their measure of local scale depends directly on the local gradient or its derivatives. Consequently, the scale cannot be measured in regions without a significant gradient, and derivative filters of larger scale have to be used in order to determine the scale there. As non-adaptive, linear filters are used to represent these larger scales, this comes down to local windows and causes blurring effects. Although these effects are often hidden behind succeeding nonlinear operators like the maximum operator, some accuracy is lost here.

* Our research is partly funded by the project WE 2602/1-1 of the *Deutsche Forschungsgemeinschaft (DFG)*. This is gratefully acknowledged. We also want to thank Mikaël Rousson and Rachid Deriche for many interesting discussions on texture segmentation.

T. Pajdla and J. Matas (Eds.): ECCV 2004, LNCS 3022, pp. 578–590, 2004.
© Springer-Verlag Berlin Heidelberg 2004

In this paper we embark on another strategy that is not edge based but region based. Our local scale measure does not depend on the behavior of the gradient in scale space, but directly on the size of regions. This poses the question of how to define regions. In order to obtain an efficient technique, some special properties of total variation (TV) denoising, discovered recently in [4], are exploited. TV denoising [18] tends to yield piecewise constant, segmentation-like results, answering the question of how to define regions. Furthermore, it holds that pixels change their value inversely proportional to the size of the region they belong to. This directly results in a region based local scale measure for each pixel that does not need the definition of any window, and therefore yields the maximum localisation accuracy.

Note that there are applications for both edge based and region based methods. In the case of scale selection for edge detection, for instance, an edge based measure makes much more sense than a region based measure. For region based texture segmentation, however, it is exactly vice-versa.

The main motivation of why a region based local scale measure is important, can be found in the field of texture discrimination. There is a consistent opinion in the literature that Gabor filters [7] yield a good vocabulary for describing textures. Furthermore, findings in neurobiology indicate that some mechanism similar to Gabor filters is used in human vision [13]. A Gabor filter bank consists of a filter for each scale and each orientation. Unfortunately, this results in a large number of features that have to be integrated e.g. in a segmentation approach. As this may cause many problems, it has been proposed to reduce this over-complete basis by the general idea of sparse coding [14]. Another approach is to avoid the high dimensionality by using nearly orthogonal measures which extract the same features from the image, namely magnitude, orientation, and scale. An early approach to this strategy has been the use of the second moment matrix for texture discrimination in [2] and [15]. More recently, a texture segmentation technique based on the features of the second moment matrix coupled with nonlinear diffusion, the so-called nonlinear structure tensor, has been presented [17]. This publication demonstrated the performance of such a reduced set of features in texture discrimination. However, the second moment matrix only holds the information of the magnitude and orientation of a structure. The information of scale is missing. Consequently, the method fails as soon as two textures can only be distinguished by means of their scale. One can expect that a local scale measure would be very useful in this respect. Deriving such a measure for texture segmentation is the topic of the present paper.

Paper organisation. In the next section the new local scale measure based on TV flow is introduced. In Section 3 this measure will be coupled with the image intensity and the second moment matrix to form a new set of texture features. For demonstrating its performance in discrimination of textures, it will be used with the segmentation approach of [17]. We show experimental results and conclude the paper with a summary.

2 Local Scale Measure

In order to obtain a region based scale measure, an aggregation method is needed that determines regions. For this purpose, we focus on a nonlinear diffusion technique, the so-called *TV flow* [1], which is the parabolic counterpart to *TV regularisation* [18].

This diffusion method tends to yield piecewise constant, segmentation-like results, so it implicitly provides the regions needed for measuring the local scale. Starting with an initial image I, the denoised and simplified version u of the image evolves under progress of artificial time t according to the partial differential equation (PDE)

$$\partial_t u = \operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right) \qquad u(t=0) = I. \tag{1}$$

The evolution of u bit by bit leads to larger regions in the image, inside which all pixels have the same value. The goal is to measure the size of these regions.
Another useful property of TV flow, besides its tendency to yield segmentation-like results, is its linear contrast reduction [4]. This allows an efficient computation of the region sizes, without explicit computation of regions. Due to the linear contrast reduction, the size of a region can be estimated by means of the evolution speed of its pixels.
In 1D, space-discrete TV flow (and TV regularisation) have been proven to comply with the following rules [4]:

(i) A region of m neighbouring pixels with the same value can be considered as one superpixel with mass m.

(ii) The evolution splits into merging events where pixels melt together to larger pixels.

(iii) Extremum pixels adapt their value to that of their neighbours with speed $\frac{2}{m}$.

(iv) The two boundary pixels adapt their value with half that speed.

(v) All other pixels do not change their value.

These rules lead to a very useful consequence: by simply sitting upon a pixel and measuring the speed with which it changes its value, it is possible to determine its local scale. As pixels belonging to small regions move faster than pixels belonging to large regions (iii), the rate of change of a pixel determines the size of the region it currently belongs to. Integrating this rate of change over the evolution time and normalising it with the evolution time T yields the average speed of the pixel, i.e. its average inverse scale in scale space.

$$\frac{1}{m} = \frac{1}{2}\frac{\int_0^T |\partial_t u|\, dt}{T} \tag{2}$$

The integration has to be stopped after some time T, otherwise the interesting scale information will be spoiled by scale estimates stemming from heavily oversimplified versions of the image. The choice of T will be discussed later in this section.
Since only extremum regions change their value, periods of image evolution in which a pixel is not part of such an extremum region have to be taken into account. This can be done by reducing the normalisation factor T by the time at which the pixel does not move (v), leading to the following formula:

$$\frac{1}{m} = \frac{1}{2}\frac{\int_0^T |\partial_t u|\, dt}{\int_0^T (1 - \delta_{\partial_t u, 0})\, dt} \tag{3}$$

where $\delta_{a,b} = 1$ if $a = b$, and 0 otherwise. Besides the estimation error for the boundary regions, where the scale is overestimated by a factor 2 (iv), this formula yields exact estimates of the region sizes without any explicit representation of the regions in the 1D case.

In 2D the topology of regions can become more complicated. A region can be an extremum in one direction and a saddle point in another direction. Therefore, it is no longer possible to obtain an *exact* estimate of the region sizes without an explicit representation of the regions. However, the extraction of regions is time consuming, and the formula in Eq. (3) still yields good approximations for the local scale in 2D, as can be seen in Fig. 1. Moreover, for texture discrimination, only a *relative* measure of the local scale is needed, and this measure will be combined with other texture features. When comparing two textured regions, the estimation error appears in both regions. For the case that the error is different in the two regions, they can be distinguished necessarily by the other texture features, since the topology of the two textures must be different. So the estimation error of the scale may not frustrate the correct distinction of two textured regions.

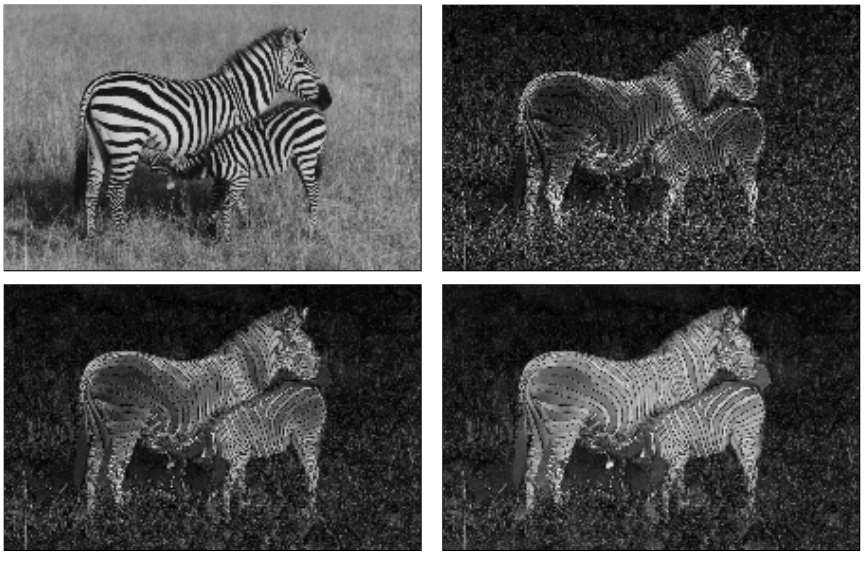


Fig. 1. TOP LEFT: (a) Zebra test image. TOP RIGHT: (b) Local scale measure with $T = 20$. BOTTOM LEFT: (c) $T = 50$. BOTTOM RIGHT: (d) $T = 100$. The scale measure yields the inverse scale, i.e. dark regions correspond to large scales, bright regions to fine scales.

On the first glance, it is surprising that there appears a scale parameter T in the scale measure. On the other hand, it is easy to imagine that a pixel lives on several different scales during the evolution of an image I from $t = 0$ to $t = T_{max}$, where the image

is simplified to its average value.[1] At the beginning, the pixel might be a single noise pixel that moves very fast until it is merged with other pixels to a small scale region. This small scale region will further move until it is merged with other regions to form a region of larger scale, and so on. Due to the integration, the average scale of the pixel's region during this evolution is measured, i.e. the complete history of the pixel up to a time T is included. For texture discrimination one is mainly interested in the scale of the small scale texture elements, so it is reasonable to emphasise the smaller scales by stopping the diffusion process before T_{max}. [2]. Fig. 1 shows the local scale measure for different parameters T. In Fig. 1d the pixels of the grass texture have been part of the large background region for such a long time, that their small scale history has hardly any influence anymore. Here, mainly the stripes of the zebras stand out from the background. On the other hand, both results shown in Fig. 1b and Fig. 1c are good scale measures for small scale texture discrimination. Note that the diffusion speed of the pixels, i.e. the *inverse* scale, is computed, so dark regions correspond to large scales. For better visibility, the values have been normalised to a range between 0 and 255.

Implementation. TV flow causes stability problems as soon as the gradient tends to zero. Therefore the PDE has to be stabilised artificially by adding a small positive constant ϵ to the gradient (e.g. $\epsilon = 0.01$).

$$\partial_t u = \operatorname{div}\left(\frac{\nabla u}{\sqrt{u_x^2 + u_y^2 + \epsilon^2}}\right) \tag{4}$$

The stability condition for the time step size τ of an explicit Euler scheme is $\tau \leq 0.25\epsilon$, so for small ϵ, many iterations are necessary. A much more efficient approach is to use a semi-implicit scheme such as AOS [20], which is unconditionally stable, so it is possible to choose $\tau = 1$. The discrete version of the scale measure for arbitrary τ is

$$\begin{aligned} u^0 &= I \\ u^{k+1} &= \frac{1}{2}\Big(\big((1) - 2\tau A_x(u^k)\big)^{-1} + \big((1) - 2\tau A_y(u^k)\big)^{-1} \Big) u^k \\ \frac{1}{m} &= \frac{1}{4\tau} \frac{\sum_{k=1}^{T} |u^{k+1} - u^k|}{\sum_{k=1}^{T} \left(1 - \delta_{(u^{k+1}-u^k),0}\right)} \end{aligned}$$

where (1) denotes the unit matrix. A_x and A_y are the diffusion matrices in x and y direction (cf. [20]). The pre-factor $\frac{1}{4}$ instead of $\frac{1}{2}$ from Eq. (3) is due to the 2D case, where a pixel has 4 neighbours instead of 2.

[1] This finite extinction time T_{max} is a special property of TV flow. An upper bound can be computed from the worst case scenario of an image with two regions of half the image size, maximum contrast c_{max}, and minimum boundary: $T_{max} \leq \frac{1}{4}\text{size}(I) \cdot c_{max}$.

[2] It would certainly be possible to automatically select an optimal scale T where only large scale object regions remain in the image [12]. However, due to the integration, the parameter T is very robust and for simplicity can be fixed at a reasonable value. For all experiments we used $T = 20$.

3 A Set of Texture Features

The local scale measure can be combined with the texture features proposed in [17]. Together, the four texture features and the image intensity can distinguish even very similar textures, as it will be shown in the next section.

Texture discrimination is a fairly difficult topic, since there is no clear definition, what texture is. Texture models can be distinguished into generative models and discriminative models. Generative models describe textures as a linear superposition of bases and allow to reconstruct the texture from its parameters. On the other side, discriminative models only try to find a set of features that allows to robustly distinguish different textures. A sound generative model has been introduced recently in [21]. A review of discriminative models can be found in [16]. While generative models are much more powerful in accurately describing textures, they are difficult to use for discrimination purposes so far.

The set of texture features we propose here, uses a discriminative texture model. It describes a texture by some of its most important properties: the intensity of the image, the magnitude of the texture, as well as its orientation and scale. These features benefit from the fact that they can be extracted also *locally*. Moreover, they can be used without further learning in any segmentation approach.

Orientation and magnitude of a texture are covered by the second moment matrix [6,15, 2]

$$J = \begin{pmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{pmatrix} \tag{5}$$

yielding three different feature channels. Furthermore, the scale is captured by our local scale measure, while the image intensity is directly available. It should be noted that in contrast to many other features used for texture discrimination, all our features are completely rotationally invariant.

As proposed in [17], a coupled edge preserving smoothing process is applied to the feature vector. This coupled smoothing deals with outliers in the data, closes structures, and synchronises all channels, which eases further processing, e.g. in a segmentation framework. For a fair coupling it is necessary that all feature channels have approximately the same dynamic range. Furthermore, the normalisation procedure must not amplify the noise in the case that one channel shows only a low contrast. Therefore, only normalisation procedures that are independent of the contrast in the input data are applicable.

As a consequence, the second moment matrix is replaced by its square root. Given the eigenvalue decomposition $J = T(\lambda_i)T^\top$ of this positive semidefinite and symmetric matrix, the square root can be computed by

$$\tilde{J} := \sqrt{J} = T(\sqrt{\lambda_i})T^\top. \tag{6}$$

Since J has eigenvalues $|\nabla I|^2$ and 0 with corresponding eigenvectors $\frac{\nabla I}{|\nabla I|}$ and $\frac{\nabla I^\perp}{|\nabla I|}$, this comes down to

$$\tilde{J} = \begin{pmatrix} \frac{I_x}{|\nabla I|} & -\frac{I_y}{|\nabla I|} \\ \frac{I_y}{|\nabla I|} & \frac{I_x}{|\nabla I|} \end{pmatrix} \begin{pmatrix} |\nabla I| & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \frac{I_x}{|\nabla I|} & \frac{I_y}{|\nabla I|} \\ -\frac{I_y}{|\nabla I|} & \frac{I_x}{|\nabla I|} \end{pmatrix} = \begin{pmatrix} \frac{I_x^2}{|\nabla I|} & \frac{I_x I_y}{|\nabla I|} \\ \frac{I_x I_y}{|\nabla I|} & \frac{I_y^2}{|\nabla I|} \end{pmatrix} = \frac{J}{|\nabla I|}. \tag{7}$$

Using one-sided differences for the gradient approximation, the components of $\tilde{J}$ have the same dynamic range as the image I. With central differences, they have to be multiplied with a factor 2.
The range of the inverse scale $\frac{1}{m}$ is between 0 and 1, so after a multiplication with 255 (the maximum value of standard grey level images) all features have values that are bounded between 0 and 255.
In [17], TV flow was proposed for the coupled smoothing of the feature vector. Here, we use TV regularisation instead. In 1D, TV flow and TV regularisation yield exactly the same output [4]. In 2D, this equivalence could not be proven so far, however, both processes at least approximate each other very well. Hence, we consider the energy functional

$$E(u) = \int_\Omega \Big((u_1 - I)^2 + (u_2 - \tilde{J}_{11})^2 + (u_3 - \tilde{J}_{22})^2 + (u_4 - 2\tilde{J}_{12})^2 + (s \cdot u_5 - r)^2 + 2\alpha \sqrt{\sum_{k=1}^{5} |\nabla u_k|^2 + \epsilon^2} \Big)\, dx$$

which consists of 5 data terms, one for each channel, and a coupled smoothness constraint that minimises the total variation of the output vector u. The values r and s are the numerator and denominator of the scale measure.

$$r := \frac{255}{4T} \int_0^T |\partial_t u|\, dt$$
$$s := \frac{1}{T} \int_0^T (1 - \delta_{\partial_t u, 0})\, dt$$

The advantage of formulating the coupled smoothing as a regularisation approach is the so-called *filling-in effect* in cases where r and s are small, i.e. the confidence in the scale measure at this pixel is small. In such a case the term $(s \cdot u_5 - r)^2$ holding the scale measure gets small, so the smoothness term with the information of the other feature channels acquires more influence. Moreover, a division by 0 is avoided, if $r = s = 0$, what can happen when a pixel has never been part of an extremum region.
The Euler-Lagrange equations of this energy are given by

$$\begin{aligned}
u_1 - I - \alpha \,\mathrm{div}\, (g \cdot \nabla u_1) &= 0 \\
u_2 - \tilde{J}_{11} - \alpha \,\mathrm{div}\, (g \cdot \nabla u_2) &= 0 \\
u_3 - \tilde{J}_{22} - \alpha \,\mathrm{div}\, (g \cdot \nabla u_3) &= 0 \\
u_4 - \tilde{J}_{12} - \alpha \,\mathrm{div}\, (g \cdot \nabla u_4) &= 0 \\
(s \cdot u_5 - r) \cdot s - \alpha \,\mathrm{div}\, (g \cdot \nabla u_5) &= 0
\end{aligned}$$

with $g := 1/\sqrt{\sum_{k=1}^{5} |\nabla u_k|^2 + \epsilon^2}$. They lead to a nonlinear system of equations, which can be solved by means of fixed point iterations and SOR in the inner loop. This has a similar efficiency as the AOS scheme used in [17] for TV flow. The texture features after the coupled smoothing are depicted in Fig. 2 and Fig. 3.

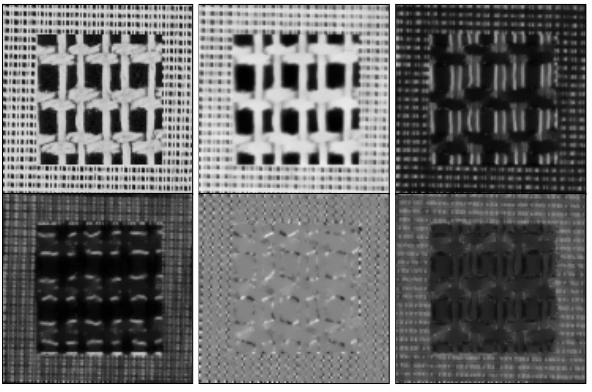


Fig. 2. FROM LEFT TO RIGHT, TOP TO BOTTOM: (a) Original image I (120×122). (b) Smoothed I ($\alpha = 46.24$). (c) Smoothed $\tilde{J}_{11}$. (d) Smoothed $\tilde{J}_{22}$. (e) Smoothed $\tilde{J}_{12}$. (f) Smoothed scale measure (inverse scale).

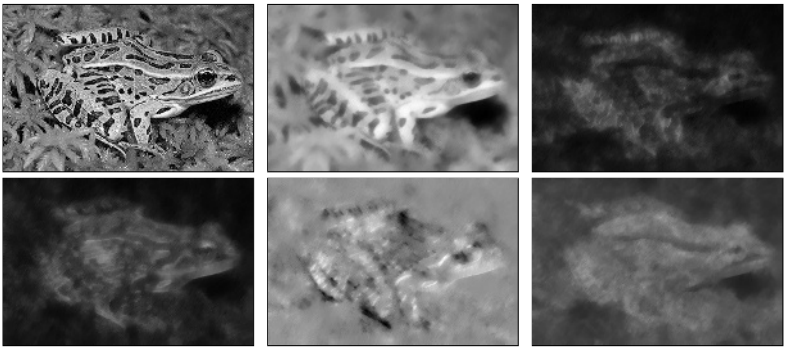


Fig. 3. FROM LEFT TO RIGHT, TOP TO BOTTOM: (a) Original image I (329×220). (b) Smoothed I ($\alpha = 189.46$). (c) Smoothed $\tilde{J}_{11}$. (d) Smoothed $\tilde{J}_{22}$. (e) Smoothed $\tilde{J}_{12}$. (f) Smoothed scale measure (inverse scale).

4 Results

Before testing the performance of the texture features in a segmentation environment, we computed the dissimilarity between several textures from the Brodatz texture database [3]. As measure of dissimilarity for each texture channel we have chosen a simple distance measure taking into account the means $\mu_k(T)$ and the standard deviations $\sigma_k(T)$ of each feature channel k of two textures T_1 and T_2:

$$\Delta_k = \left(\frac{\mu_k(T_1) - \mu_k(T_2)}{\sigma_k(T_1) + \sigma_k(T_2)} \right)^2 . \tag{8}$$

For the total dissimilarity the average of all 5 texture channels is computed:

$$\Delta = \frac{1}{5} \sum_{k=1}^{5} \Delta_k. \tag{9}$$

The resulting dissimilarities are shown in Fig. 4. The first value in each cell is the dissimilarity Δ_5 according to the local scale measure only. The second value is the dissimilarity Δ taking all texture features into account. The computed values are in accordance with what one would expect from a measure of texture dissimilarity. Note that there are cases where Δ_5 is significantly larger than Δ. In such cases the local scale measure is very important to reliably distinguish the two textures.

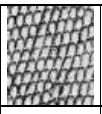
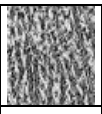
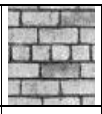
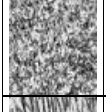
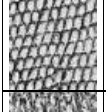
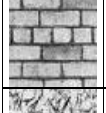
								
	0 **0**	147 **114**	399 **865**	1388 **1514**	1049 **379**	23 **130**	4449 **1893**	735 **663**
	147 **114**	0 **0**	92 **663**	862 **1656**	583 **470**	60 **204**	4174 **2258**	331 **456**
	399 **865**	92 **663**	0 **0**	340 **3068**	199 **899**	269 **565**	2870 **3630**	74 **846**
	1388 **1514**	862 **1656**	340 **3068**	0 **0**	11 **716**	1235 **1443**	1546 **421**	78 **538**
	1049 **379**	583 **470**	199 **899**	11 **716**	0 **0**	897 **274**	1589 **923**	27 **454**
	23 **130**	60 **204**	269 **565**	1235 **1443**	897 **274**	0 **0**	4548 **2000**	589 **663**
	4449 **1893**	4174 **2258**	2870 **3630**	1546 **2107**	1589 **923**	4548 **2000**	0 **0**	1932 **983**
	735 **663**	331 **456**	74 **846**	78 **538**	27 **454**	589 **663**	1932 **983**	0 **0**

Fig. 4. Dissimilarities measured between some Brodatz textures. First Value: Dissimilarity according to scale measure. Second Value: Dissimilarity according to all texture features.

It should be noted that in this simple experiment there have been no boundaries between textures, as the statistics have been computed separately for each texture. Therefore, this experiment shows only the global texture discrimination capabilities of the features,

when the texture regions are already known. In order to test the performance of the features when being applied to the much harder problem of discriminating textures with the region boundaries not known in advance, the features were incorporated into the segmentation technique described in [17]. Here, the good localisation accuracy of the features becomes crucial.

First the segmentation was applied to images which were chosen such that they mainly differ in scale. The results are depicted in Fig. 6 and Fig. 7. Note that the inner region of Fig. 7 is the downsampled version of the outer region so the regions solely differ in scale. Therefore, the segmentation fails, if the feature channel representing our scale measure is left out. It also fails, if this channel is replaced by three channels with responses of circular Gabor filters of different scale (see Fig. 5).

This shows, at least for texture segmentation, that the use of a direct scale measure outperforms the representation of scale by responses of a filter bank. Note that our scale measure is also more efficient, as the segmentation has to deal with less channels. The total computation time for the 220×140 zebra image on an Athlon XP 1800+ was 8 seconds, around 0.5 seconds to extract the local scale, 1.5 seconds to regularise the feature space, and 6 seconds for the segmentation.

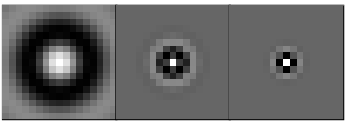


Fig. 5. Filter masks of the three circular Gabors used in the experiments.

Finally, the method was tested with two very challenging real world images. The results are depicted in Fig. 8 and Fig. 9. The fact that the method can handle the difficult frog image, demonstrates the performance of the extracted features used for the segmentation. Also the very difficult squirrel image, which was recently used in [8], could be segmented correctly, besides a small perturbation near the tail of the squirrel. Here again, it turned out that our scale representation compares favourably to circular Gabor filters.

5 Summary

In this paper, we presented a region based local scale measure. A diffusion based aggregation method (TV flow) is used to compute a scale space representing regions at different levels of aggregation. By exploiting the linear contrast reduction property of TV flow, the size of regions can be approximated efficiently without an explicit representation of regions. For each pixel and each level of aggregation the scale is determined by the size of the region the pixel belongs to. This scale is integrated over the diffusion time in order to yield the average scale for each pixel in scale space.

The local scale measure has been combined with other texture features obtained from the second moment matrix and the image intensity. Together, these features cover the most important discriminative properties of a texture, namely intensity, magnitude, orientation, and scale with only 5 feature channels. Consequently, a supervised learning stage to reduce the number of features is not necessary, and the features can be used directly with any segmentation technique that can handle vector-valued data. The performance in

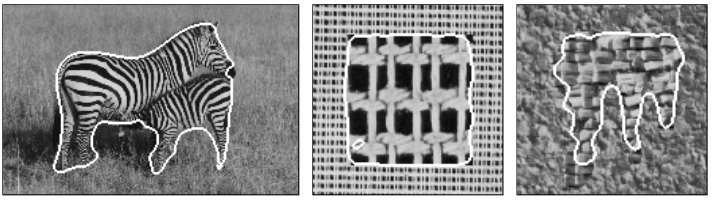


Fig. 6. Segmentation with the proposed texture features.

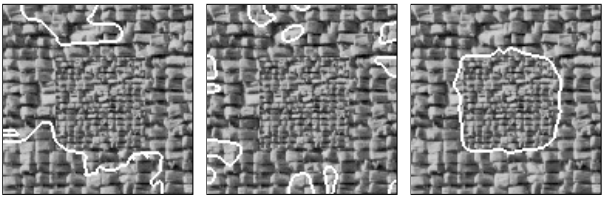


Fig. 7. LEFT: Segmentation without a scale measure. CENTER: Segmentation using 3 circular Gabors for scale. RIGHT: Segmentation using our local scale measure instead.

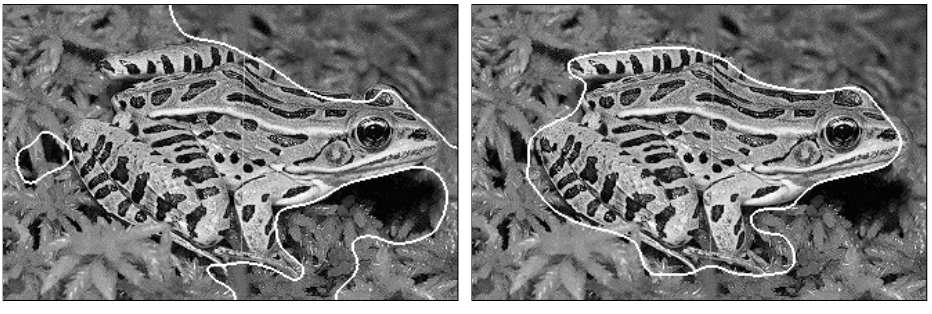


Fig. 8. LEFT: Segmentation using 3 circular Gabors for scale. RIGHT: Segmentation using our local scale measure instead.

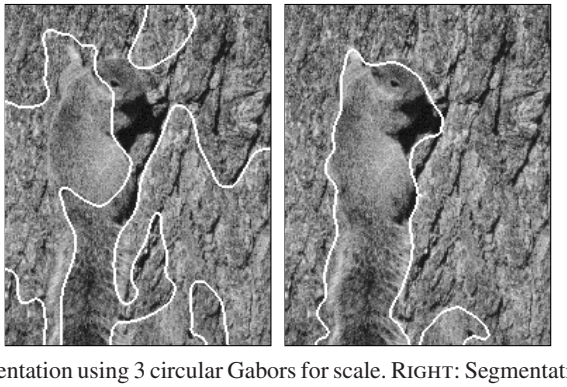


Fig. 9. LEFT: Segmentation using 3 circular Gabors for scale. RIGHT: Segmentation using our local scale measure instead.

texture discrimination has successfully been demonstrated by the segmentation of some very difficult texture images.

References

1. F. Andreu, C. Ballester, V. Caselles, and J. M. Mazón. Minimizing total variation flow. *Differential and Integral Equations*, 14(3):321–360, Mar. 2001.
2. J. Bigün, G. H. Granlund, and J. Wiklund. Multidimensional orientation estimation with applications to texture analysis and optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8):775–790, Aug. 1991.
3. P. Brodatz. *Textures: a Photographic Album for Artists and Designers*. Dover, New York, 1966.
4. T. Brox, M. Welk, G. Steidl, and J. Weickert. Equivalence results for TV diffusion and TV regularisation. In L. D. Griffin and M. Lillholm, editors, *Scale-Space Methods in Computer Vision*, volume 2695 of *Lecture Notes in Computer Science*, pages 86–100. Springer, Berlin, June 2003.
5. J. H. Elder and S. W. Zucker. Local scale control for edge detection and blur estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(7):699–716, July 1998.
6. W. Förstner and E. Gülch. A fast operator for detection and precise location of distinct points, corners and centres of circular features. In *Proc. ISPRS Intercommission Conference on Fast Processing of Photogrammetric Data*, pages 281–305, Interlaken, Switzerland, June 1987.
7. D. Gabor. Theory of communication. *Journal IEEE*, 93:429–459, 1946.
8. M. Galun, E. Sharon, R. Basri, and A. Brandt. Texture segmentation by multiscale aggregation of filter responses and shape elements. In *Proc. IEEE International Conference on Computer Vision*, Nice, France, Oct. 2003. To appear.
9. G. Gómez, J. L. Marroquín, and L. E. Sucar. Probabilistic estimation of local scale. In *Proc. International Conference on Pattern Recognition*, volume 3, pages 798–801, Barcelona, Spain, Sept. 2000.
10. H. Jeong and I. Kim. Adaptive determination of filter scales for edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(5):579–585, May 1992.
11. T. Lindeberg. *Scale-Space Theory in Computer Vision*. Kluwer, Boston, 1994.
12. T. Lindeberg. Principles for automatic scale selection. In B. Jähne, H. Haußecker, and P. Geißler, editors, *Handbook on Computer Vision and Applications*, volume 2, pages 239–274. Academic Press, Boston, USA, 1999.
13. S. Marcelja. Mathematical description of the response of simple cortical cells. *Journal of Optical Society of America*, 70:1297–1300, 1980.
14. B. A. Olshausen and D. J. Field. Sparse coding with an over-complete basis set: A strategy employed by V1? *Vision Research*, 37:3311–3325, 1997.
15. A. R. Rao and B. G. Schunck. Computing oriented texture fields. *CVGIP: Graphical Models and Image Processing*, 53:157–185, 1991.
16. T. R. Reed and J. M. H. du Buf. A review of recent texture segmentation and feature extraction techniques. *Computer Vision, Graphics and Image Processing*, 57(3):359–372, May 1993.
17. M. Rousson, T. Brox, and R. Deriche. Active unsupervised texture segmentation on a diffusion based feature space. In *Proc. 2003 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, volume 2, pages 699–704, Madison, WI, June 2003. IEEE Computer Society Press.
18. L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.
19. J. Sporring, C. I. Colios, and P. E. Trahanias. Generalized scale-selection. Technical Report 264, Foundation for Research and Technology - Hellas, Crete, Greece, Dec. 1999.

20. J. Weickert, B. M. ter Haar Romeny, and M. A. Viergever. Efficient and reliable schemes for nonlinear diffusion filtering. *IEEE Transactions on Image Processing*, 7(3):398–410, Mar. 1998.
21. S.-C. Zhu, C. Guo, Y. Wu, and W. Wang. What are textons? In A. Heyden, G. Sparr, M. NIelsen, and P. Johansen, editors, *Proc. 7th European Conference on Computer Vision*, volume 2353 of *Lecture Notes in Computer Science*, pages 793–807, Copenhagen, Denmark, May 2002. Springer.