Gradient-Based Segmentation Requires Doubling of the Sampling Rate

Ullrich Köthe
University of Hamburg, Department of Informatics
Email: koethe@informatik.uni-hamburg.de

Abstract

We apply Shannon's sampling theorem to gradient based image segmentation and show that the gradient magnitude requires twice the sampling rate of the original image. Taking this simple, but apparently overlooked phenomenon into account, we demonstrate experimentally that correct sampling of the gradient image indeed improves the quality of the resulting segmentation.

1. Introduction

The gradient is one of the most important tools in low-level image analysis. For example, gradient calculations are at the heart of Canny's edge detector [1] and the corner response function [3]. Therefore, a better understanding of the properties of gradient operators would be of great benefit for the improvement of gradient-based algorithms and for the accurate prediction of algorithm performance in a particular application.

One area where gradient operators are poorly understood is their interaction with the process of sampling. Most gradient-based algorithms are derived in the analog domain, and discrete algorithm implementations on a computer are simply considered as approximations of the analog theory. But in most cases little is known about how accurate the approximation really is. Most theoretical investigations of approximation quality only yield asymptotic results which tell how fast the accuracy improves as the sampling rate is increased. However, in image analysis the sampling rate cannot in general be made arbitrarily high – it is more or less fixed once the imaging conditions (the properties of the camera and the geometric relationships between the camera and the scene) have been defined. Therefore, in image analysis we need absolute bounds for the question of whether something will be visible at a given resolution.

One instance of such an absolute measure of accuracy is *Shannon's sampling theorem*, see e.g. [2]. The sampling theorem states that any band-limited signal can be exactly resonstructed from a countable number of samples. That is, when the signal's frequency spectrum is zero above a certain limit frequency, no information is lost by sampling, provided the sample distance does not exceed the Nyquist distance calculated from the limit frequency. Since any real data channel, and in particular any real camera, is band-limited, the sampling theorem certainly applies to image analysis problems. Now it is interesting to ask whether actual image analysis algorithms actually adhere to the constraints set by the theorem. In case of gradient-based segmentation, there is some anecdotal evidence to the contrary: several people we spoke to have observed the phenomenon that segmentation results on some images improved when these images were interpolated to a higher resolution before analysis. This suggests that the original sampling density was too low. It should be stressed that the improvements were achieved by interpolation of the given images, not by taking new images at higher resolution. Unfortunately, such results are rarely published. Overington's book [5] is the only definitive exception we are aware of – it shows convincing examples of improved edge detection in interpolated images.

In this paper, we put forward a simple signal-theoretic argument that this phenomenon has indeed a theoretical explanation. The analysis suggests that a doubling of the resolution is actually necessary to obtain correct gradient measurements (unless the original image doesn't contain high frequencies in the first place). We show that the oversampling is best realized directly by the gradient filter, without a dedicated interpolation step. Experiments demonstrate that this modification indeed leads to improved segmentation results.

2. Sampling Limits for the Gradient Magnitude

Consider a band-limited image f that we want to analyse. To be band-limited means that the Fourier spectrum of the image is zero above some spatial frequency Ω_0:

$$\mathcal{F}[f(\vec{x})] = F(\vec{\omega}) = 0, \text{ if } |\vec{\omega}| \geq \Omega_0 \quad (1)$$

where $\mathcal{F}$ denotes the Fourier transform. The sampling theorem now states that a band-limit signal must be sampled with a sampling frequency of at least $2\Omega_0$. This minimum sampling frequency is known as the *Nyquist frequency*. By means of the derivative theorem of Fourier theory we can express the image's first derivative in horizontal direction f_{x_1} in the Fourier domain as follows (the vertical derivative is treated analogously):

$$\begin{aligned} f_{x_1}(\vec{x}) &= \frac{d}{d\,x_1} f(\vec{x}) \\ \Rightarrow \quad \mathcal{F}[f_{x_1}](\vec{\omega}) &= -j\omega_1 F(\vec{\omega}) \end{aligned} \tag{2}$$

where x_1, ω_1 denote the horizontal components of the coordinates in the spatial and Fourier domains respectively. It can thus be seen that taking the derivative does not change the support region of the Fourier transform – the derivative image remains band-limited with band-limit Ω_0. In image processing, the horizontal derivative is usually calculated by means of a suitable derivative filter k_{x_1}. Spatial convolution of the image with the derivative filter amounts to multiplication of the image spectrum with the kernel's Fourier transform:

$$\begin{aligned} f_{x_1}(\vec{x}) &= (k_{x_1} \star f)(\vec{x}) \\ \Rightarrow \quad \mathcal{F}[f_{x_1}](\vec{\omega}) &= K_{x_1}(\vec{\omega}) F(\vec{\omega}) \end{aligned} \tag{3}$$

with $K_{x_1} = \mathcal{F}[k_{x_1}]$. The band limit of the convolution equals the minimum of the band limits of the two factors. Since all practical relevant derivative filters (finite differences, derivatives of Gaussian filters and, generally, all finite impulse response filters) are not band-limited, the thus obtained derivative image has again band-limit Ω_0. Now, most algorithms using first derivatives need to calculate the gradient squared magnitude $f_{x_1}^2 + f_{x_2}^2$. The multiplication of a signal with itself in the spatial domain corresponds to a convolution of the spectrum with itself in the Fourier domain:

$$\mathcal{F}[f_{x_1}^2] = \mathcal{F}[f_{x_1}] \star \mathcal{F}[f_{x_1}] \tag{4}$$

If $\chi[f]$ denotes the support region of a function, then the support region of a convolution result can be calculated as the morphological dilation:

$$\chi[f \star g] = \chi[f] \oplus \chi[g] \tag{5}$$

($\oplus$ is the dilation operator). Applying this to our problem, we see that taking the square of f_{x_1} doubles the limit frequency:

$$\mathcal{F}[f_{x_1}^2](\vec{\omega}) = 0, \text{ if } |\vec{\omega}| \geq \Omega_1 = 2\Omega_0 \tag{6}$$

Consequently, the Nyquist frequency must also double. In order to correctly represent the gradient squared magnitude, we *must halve the sample distance*. Taking the square root (i.e. calculating the gradient magnitude) does not change this in any essential way. The following 1-dimensional example demonstrates that this analysis is indeed correct (compare fig. 1). Let the original signal be given by

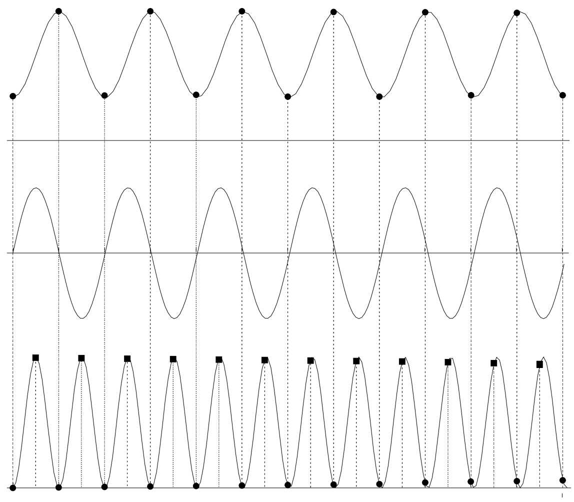

Figure 1: Top: sine-wave $\sin(\omega x + \tau_0) + c_0$, sampled at the rate $\lambda_0 = \frac{\pi}{\omega+\epsilon}$ (black balls); center: analytical first derivative of the sine-wave; bottom: squared first derivative, sampled at integer locations (balls) and half-integer locations (squares).

$$f(x) = \sin(\omega x + \tau_0) + c_0 \tag{7}$$

where τ_0 is the phase angle and c_0 the DC component. This signal is band-limited with $\Omega_0 = \omega + \epsilon$ for arbitrary small positive ϵ. It can be correctly sampled at the Nyquist rate $\lambda_0 = \frac{\pi}{\Omega_0}$ (black balls in fig. 1 top). The squared first derivative of the signal is

$$\left(\frac{d}{d\,x} f(x)\right)^2 = \frac{\omega^2}{2}(1 + \cos(2\omega x + 2\tau_0)) \tag{8}$$

Its frequency is twice the original frequency. When this function is just sampled at the original sampling positions, its character is completely lost (balls in fig. 1 bottom). To represent the signal correctly, we must additionally sample at half-integer position (squares in fig. 1 bottom) – only then do we see an oscillation of the right frequency.

For an arbitrary band-limited image, the necessity of oversampling could only be avoided if the derivative filter's band-limit were $\Omega_0/2$ or below. While no practically important filter is band-limited in theory, a rapidly decaying filter like the Gaussian

$$g_\sigma(\vec{x}) = \frac{1}{2\pi\sigma^2} e^{-\frac{\vec{x}^T\vec{x}}{2\sigma^2}} \tag{9}$$

can be considered as band-limited for all practical purposes. If we require the spectrum to be damped to 1% of its original amplitude at $\omega' = \Omega_0/2$, the scale σ of the Gaussian derivative must be at least about $2\lambda_0$, where $\lambda_0 = \frac{\pi}{\Omega_0}$ is the pixel distance of the original image. For many applications (whenever any fine detail has to be detected) this scale is too coarse – typically, edge detection is done with $\sigma \leq \lambda_0$. Then, oversampling is required.

3. Oversampling of the Gradient: Algorithm and Examples

We have seen that the correct representation of the gradient magnitude requires to double the sample density. When the gradient is calculated by means of derivative filters, as is common in image analysis, this oversampling is best realized during gradient calculation itself. A separate interpolation step can thus be avoided. Consider the definition of the convolution sum for a digital signal:

$$(f \star k)(x) = \sum_{m=-\infty}^{\infty} k(x-m) f(m) \qquad (10)$$

Since the original function f is discrete, it is only accessed at the integer coordinates m. Usually, a discrete result image $(f \star k)$ with the same sample positions is obtained by letting x also run over the integers. However, as long as the kernel k is defined at all locations $x - m$, the sum can be evaluated for *arbitrary* x. Thus, to achieve twofold oversampling, we can simply calculate $(f \star k)$ for the half-integers as well. This means that the kernel has to be defined at integers and half-integers. When continously defined kernels such as the first derivatives of Gaussians are used, this is no problem. To define a finite difference filter, we can combine the forward and the symmetric differences into one kernel. The oversampling finite difference kernel then is

$$D_{x_1} = \left(\frac{1}{2}\,,\, 1\,,\, 0\,,\, -1\,,\, -\frac{1}{2} \right) \qquad (11)$$

where the entries' coordinates are given by $(-1, -\frac{1}{2}, 0, \frac{1}{2}, 1)$, and the kernel is zero everywhere else.

In the following we demonstrate with a number of example images that gradient oversampling really yields visible improvements to the segmentation quality. We have chosen to use Canny's algorithm [1] for all examples. Of course, in any particular case a different algorithm might be more appropriate. But by sticking to a single algorithm we ensure that the examples remain somewhat comparable.

The test images were selected so that they contain interesting fine scale structure, and the contrast and signal-to-noise ratios are so good that low contrast and noise are effectively ruled out as sufficient explanations for bad segmentation results. In all cases, the gradient was calculated with Gaussian derivative filters at scale $\sigma = 0.7\lambda_0$, where λ_0 is the pixel distance in the *original* image. Thresholds were manually optimized for each example, and the same parameters were then used for both resolutions (the normalization of the derivative filters ensures that this is actually correct).

Figure 2: Left: original image; top right: segmentation of license plate at the original resolution; bottom right: the same region segmented at twice the original resolution.

The improvement is perhaps most pronounced in text recognition. Fig. 2 shows an example from license plate reading, fig. 3 one from optical character recognition. In both cases the characters can barely be recognized in the edge image with the original resolution, but the text is clearly readable at the improved resolution. Figures 4 and 5 demonstrate the application of the new method to outdoor photographs. It can be seen that the geometric structure of the fine details (e.g. window crossbars in fig. 4, the fine grating in fig. 5) is much better resolved in the oversampled image. Perhaps more importantly, the errors in the representation of these details at the original resolution have bad effects on nearby structure: although some objects are big enough to be resolved without oversampling, their representation is severely disturbed by the errors in the neighborhood. So one should use the higher resolution even when fine detail is of no interest, just to make sure that errors don't propagate into the structures one wants to analyse. Finally, fig. 6 shows an example where the structure changes its size across the image. As the size gets smaller, the segmentation at the original resolution is becoming worse to the point of being useless, whereas the segmentation at higher resolution remains correct over the entire size range. This could be of high value for shape from texture algorithms.

4. Conclusions

In this paper we showed that the gradient magnitude must in general be sampled at twice the original sampling rate. Although the theoretical explanation is simple, it seems that this phenomenon has been overlooked so far. But our examples clearly demonstrate that higher sampling rates can lead to visible improvements in the segmentation quality. The example images were selected so that noise and low contrast cannot serve as an explanation for unsatisfactory segmentation results. Given our findings we suppose that many segmentation problems which were previously attributed to noise and low contrast are actually caused by insufficient

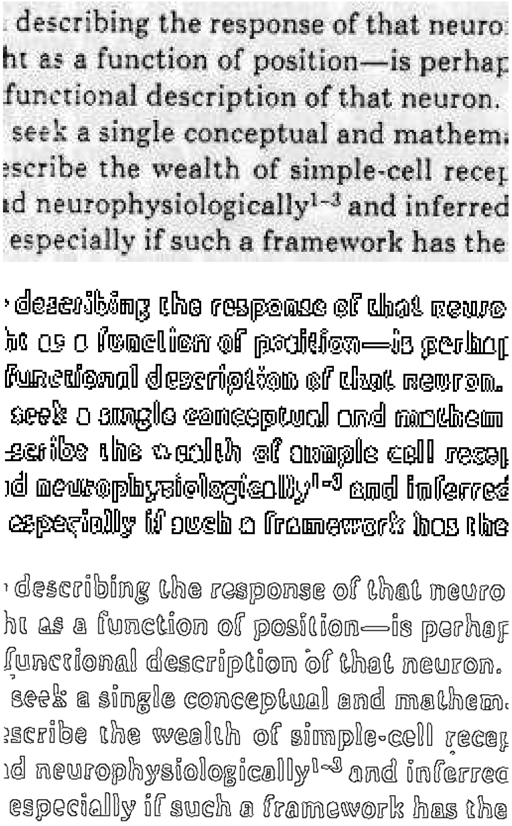

Figure 3: Top: original image; center: segmentation at the original resolution; bottom: segmentation after twofold oversampling.

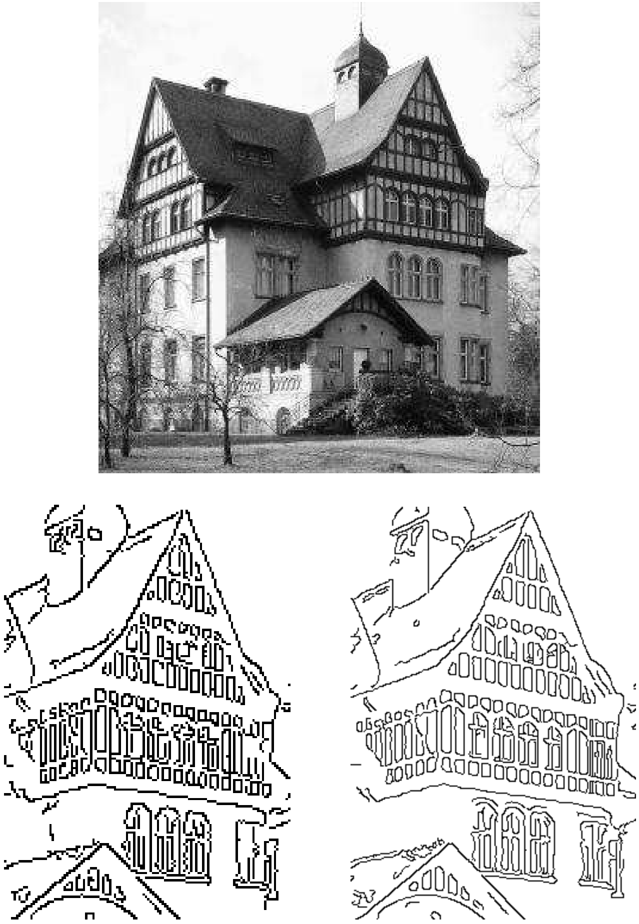

Figure 4: Top: original image; bottom left: segmentation at the original resolution; bottom right: segmentation after twofold oversampling. Especially note the differences in the windows.

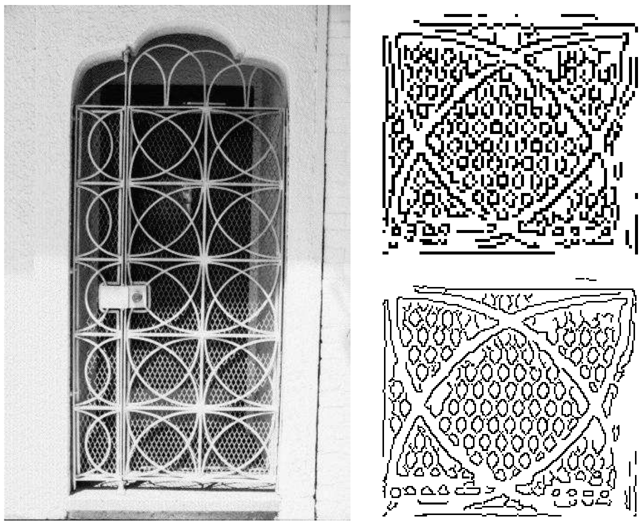

Figure 5: Left: original image; top right: detail (bottom left panel) of the segmentation at the original resolution; bottom right: segmentation of the same detail after twofold oversampling.

sampling. Although we demonstrated our results for gradient based edge detection using Canny's algorithm, the requirements on sampling apply equally to any other method that involves products of filter results.

The negligence of the sampling problem might have a very simple explanation: Until very recently, the high amount of data to be processed during image analysis required a reduction of the image size (e.g. by means of an image pyramid) rather than an increase. Fortunately, hardware capabilities are rapidly improving, so that oversampling becomes a feasible option. The necessary changes to existing methods are straightforward: Since oversampling can be done directly by the gradient filter, a system's gradient calculation module can simply be replaced with an improved one. There is no need for conceptual changes of the segmentation methods, at least as far as the findings of this paper are concerned.

While the improvement in segmentation quality is obvious, it is also clear that the results are still not perfect. Besides the well-known problems caused by noise and low contrast, the question arises whether other unexplored phenomena influence the required sampling rate. The property of being band limited does not directly tell anything about

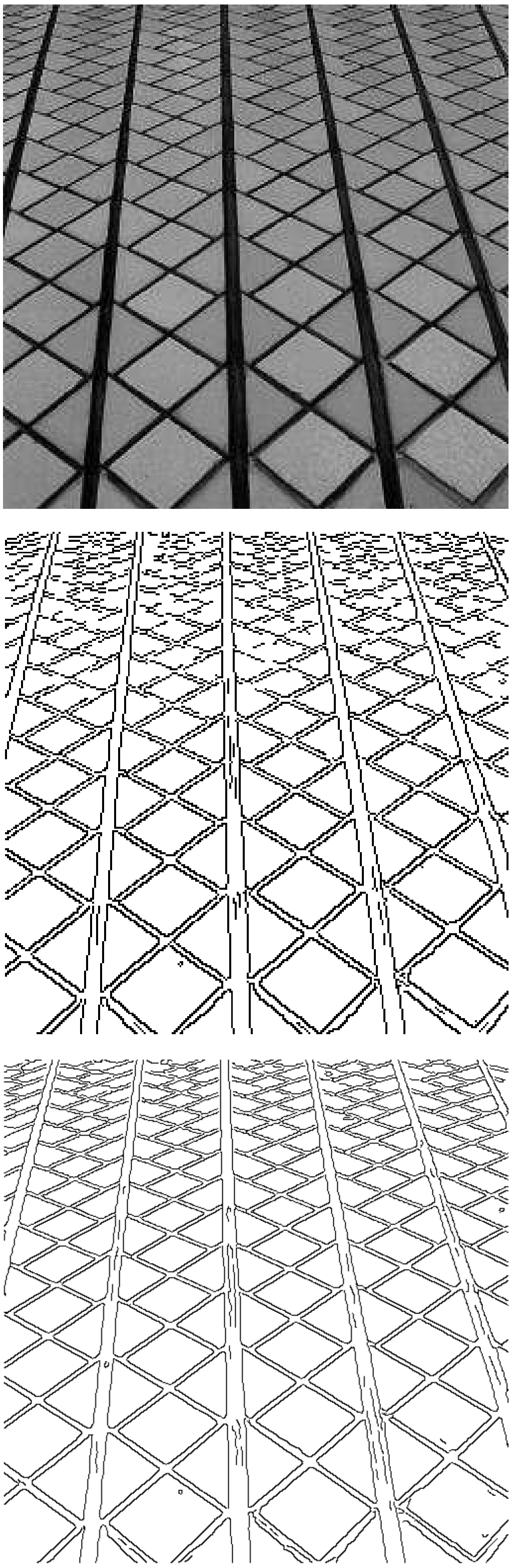

Figure 6: Top: original image; center: segmentation at the original resolution; bottom: segmentation after twofold oversampling.

the geometric content of the image. Even if an image fulfills Shannon's theorem, it can still be difficult to actually find and represent edges in such a way that the topological and geometrical structure of the resulting segmentation is correct. Unfortunately, the existing topological sampling theorems, e.g. [4, 6], do only apply to binary images. It would be interesting to find out wheter a topological sampling theorem for gray scale images exists and to check whether the bounds derived in this paper have to be tightened even more.

References

[1] J. Canny: *A Computational Approach to Edge Detection*, IEEE Trans. Pattern Analysis and Machine Intelligence, 8(6), pp. 679-698, 1986

[2] G. Granlund, H. Knutsson: *Signal Processing for Computer Vision*, Kluwer, 1995

[3] C.G. Harris, M.J. Stevens: *A Combined Corner and Edge Detector*, Proc. of 4th Alvey Vision Conference, 1988

[4] L.J. Latecki, C. Conrad, A. Gross: *Preserving Topology by a Digitization Process*, J. Mathematical Imaging and Vision, 8:131-159, 1998

[5] I. Overington: *Computer Vision – A unified, biologically-inspired approach*, Elsevier, 1992

[6] T. Pavlidis: *Algorithms for Graphics and Image Processing*, Computer Science Press, 1982