IMAGE ANALYSIS USING SOFT HISTOGRAMS

Per-Erik Forssén

Computer Vision Laboratory
Department of Electrical Engineering
Linköping University, SE-581 83 Linköping, Sweden

ABSTRACT

This paper advocates the use of overlapping bins in histogram creation. It is shown how conventional histogram creation has an inherent quantisation that cause errors much like those in sampling with insufficient band limitation. The use of overlapping bins is shown to be the deterministic equivalent to dithering. Two applications of soft histograms are shown: Improved peak localisation in an estimated probability density function (PDF) without requiring more samples, and accurate estimation of image rotation.

1. INTRODUCTION

The purpose of a histogram is to estimate how the values of a variable is distributed across a certain range, i.e. to estimate a probability density function (PDF). The most common use of histograms in image analysis is estimation of the intensity distribution.

When computing a conventional histogram, the range of values for the data is separated into a set of disjoint *bins*. For each bin one counts the number of samples that fall into its range. If we call the bin centres m_k, and the bin distance (and bin width) d, the histogram value for bin number k can be written as:

$$h_k = \sum_{n=1}^{N} H_k(s_n) \quad \text{where} \tag{1}$$

$$H_k(s_n) = \begin{cases} 1 & \text{if } |s_n - m_k| < d/2 \\ 0 & \text{otherwise} \end{cases}$$

Here s_n are samples of the variable under study, and N is the number of samples of this variable. The histogram creation procedure can be seen as an initial quantisation of the samples s_n, followed by a summation. Unless the variable under study is already quantised (as is normally the case for image intensities), the histogram creation introduces an effect similar to aliasing. We can see this by viewing the histogram creation as a band limitation of the PDF, followed by a sampling. The equivalent of a band-limitation function is $H_k(s)$, which corresponds to a $\mathrm{sinc}()$ in the Fourier domain.

The fact that the above described histogram creation in some sense violates the sampling theorem limits the uses of a histogram. The purpose of this paper is to describe a method to generate more useful histograms, and to illustrate some ways they can be put to use.

2. DITHERING

Since the histogram creation process contains an inherent quantisation, it could probably benefit from *dithering* [1]. Dithering is the process of adding a small amount of noise (with certain characteristics) to a signal prior to the quantisation. Dithering is commonly used in image reproduction with a small number of available intensities or colours, as well as in quality improvement of digital audio [2].

The initial probability for a sample to fall into a certain bin is 1 inside the bin interval (see figure 1, left). However, if we add triangular noise, or *TPDF noise*[1] (see figure 1, right) we end up with *stochastic bins*, with PDFs that are smooth, and slightly overlapping (see figure 1, centre).

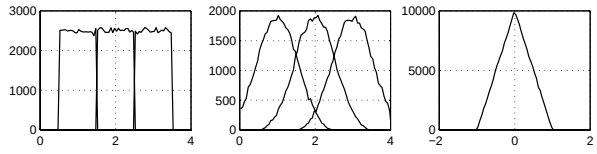

Fig. 1. Stochastic bins.
Left: *Estimated PDFs of bins* $1 \ldots 3$.
Centre: *PDFs with noise added before quantisation.*
Right: *Estimated PDF of added noise.*

The author wants to acknowledge the financial support of WITAS, the Wallenberg laboratory for Information Technology and Autonomous Systems.

[1] TPDF noise is designed to de-correlate the power spectrum of the quantisation error and that of the signal [2]. It can be generated by summation of two uniformly distributed random variables.

3. OVERLAPPING BINS

It is interesting to note that the shape of the stochastic bins in figure 1 shows a strong resemblance to the $\cos^2()$ envelope functions used in the *channel representation* [3, 4]. However, they are not quite identical. The theoretical shape of the PDF is that of a rectangular PDF convolved with a triangular one, and this shape can be represented using piecewise second order polynomials. They are however of similar shape, and are both spatially limited.

In quantisation we are usually forced to choose one bin, due to the compact representations of numbers. For such situations, changing the bins into stochastic bins is a good idea, since on the average we will get overlapping bins, and thus reduced amount of "aliasing".

However, since we are now creating histograms, we could just as well generate deterministic, but overlapping bins. For this purpose we will employ the channel representation, and compute what is called *soft histograms*.

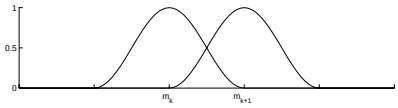

Fig. 2. Overlapping "bins".

Instead of letting each sample fall into one of the bins, we will allow it to fall into two or more neighbouring bins, but only partially (see figure 2). Using the same notation as in equation 1, the histogram value for bin (or rather channel) number k is now written as:

$$h_k = \sum_{n=1}^{N} \psi_k(s_n) \quad \text{where} \tag{2}$$

$$\psi_k(s_n) = \begin{cases} \cos^2\left(\frac{\omega}{d}(s_n - m_k)\right) & \text{if } |s_n - m_k| < \frac{\pi d}{2\omega} \\ 0 & \text{otherwise} \end{cases}$$

The parameter ω is called the *channel overlap*, and controls the correlation between the bins. Figure 2 shows the case of $\omega = \pi/2$. In practise values of $\omega = \pi/q$ where $q = 2, 3, 4 \ldots$ should be used, since these will result in constant sums of the contributions.

This approach falls into the class of *kernel density estimators* [5, 6] developed in the late 50's and early 60's by Rosenblatt and Parzen. Parzen prescribes a general kernel function instead of the rectangular function $\mathrm{H}_k(s)$ in equation 1.

Note that, contrary to the conventional approach we have not destroyed any information when letting s_n fall into two bins. We can see this by reconstructing the sample s_n from two non-zero $\psi_k(s_n)$ contributions. If the two bins with non-zero contributions are denoted k, and $k+1$, the difference between their contributions, when $\omega = \pi/2$ can be written as:

$$\psi_k(s_n) - \psi_{k+1}(s_n) = \cos\left(\frac{\pi}{d}(s_n - m_k)\right)$$

using the fact that $m_{k+1} = m_k + w$, and the well known relation $\cos(2\alpha) = cos^2(\alpha) - sin^2(\alpha)$. The sample value can thus be reconstructed as:

$$s_n = m_k + \frac{d}{\pi}\arccos(\psi_k(s_n) - \psi_{k+1}(s_n)) \tag{3}$$

As soon as we start to use more than one sample in equation 2 however, there is no way back.

If we already have computed a full conventional histogram, with bins $h_1, h_2, \ldots h_K$, and corresponding bin centres $m_1, m_2, \ldots m_K$, it can easily be converted into a soft histogram with bins $c_1, c_2, \ldots c_L$, according to:

$$c_l = \sum_{k=1}^{K} h_k \psi_l(m_k) \tag{4}$$

That is, we compute the value of each envelope function, $\psi_l(x)$, once for each bin in the conventional histogram, and multiply the result with the number of times this bin was visited. This is normally much faster than a full soft histogram computation, since the number of bins in the full histogram, K, usually is much smaller than the number of samples, N. If we ensure that K is significantly larger than L, the quantisation effects in the original histogram will be negligible.

4. ALIASING IN CONVENTIONAL HISTOGRAMS

Figure 3 illustrates the "aliasing effect" of conventional histogram computation. Both graphs show 10 superimposed histograms (with 50 bins each) that only differ in construction by a displacement. As can be seen, the soft histogram varies smoothly with the bin-centre displacement, while conventional histograms exhibit abrupt jumps in the bin values. This effect is quite similar to what happens during sampling with insufficient band-limitation.

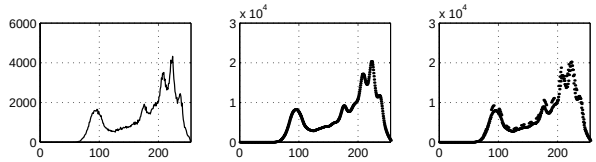

Fig. 3. Comparison of conventional and soft histograms.
Left: *Image histogram with 256 bins.*
Centre: 10 *superimposed soft histograms with* 50 *bins each.*
Right: 10 *superimposed conventional histograms with* 50 *bins each.*

The extent of the aliasing effects depend highly on the characteristics of the underlying distribution. Cases where the effect is either smaller or more severe than shown in figure 3 are easy to find.

5. FINDING PEAKS IN A SOFT HISTOGRAM

In many applications it is of interest to know at which values we have local peaks in a distribution. Due to the aliasing-like effects in conventional histograms, a local peak in the histogram does not necessarily correspond to a peak in the distribution. Provided that we have detected a peak, m_k, in the histogram, we can only say that it is *likely* that the PDF peak lies somewhere in the range $[m_k - d/2, m_k + d/2]$. If we were using conventional histograms, we would have to reduce the bin sizes and increase the number of samples in order to improve the accuracy, something which is not always possible.

With soft histograms however, the location of the peaks can be found with an accuracy that is higher than the bin distance. The bin distance only limits how close two peaks may be. If they are too close they will tend to interfere, and eventually their average will be found instead.

In order to find local peaks, we have to consider at least three consecutive bin values. We can find the peak by modelling a local interval of the PDF as a second order polynomial:

$$h = l_1 s^2 + l_2 s + l_3 \tag{5}$$

The coefficients l_1, l_2 and l_3 can be expressed as product sums of three consecutive bin values. The local peak can be found as the zero crossing of the derivative, i.e. $m = -l_2/2l_1$. If we limit our search to bin triplets where the middle bin is larger than its neighbours, we know that this is a maximum.

Figure 4 illustrates the application of this peak detection scheme to soft histograms with 13, 25, 50, and 100 bins respectively. Note how the peaks near $s = 215$ are seen as one peak in the first histogram, and gradually change into three peaks in the last one.

The accuracy of the result is of course highly dependent on the number of samples, N. Given the same set of samples, the accuracy should however always be at least as good as that of a conventional histogram.

6. SOFT HISTOGRAMS OF VECTOR FIELDS

An other application of soft histograms is detection of image rotation. This can be done using a property that is equivariant with rotation, such as local orientation [7, 8], or the DIV symmetry features [7, 9]. In this paper we will illustrate the method using soft histograms of local orientation, but DIV features work just as well.

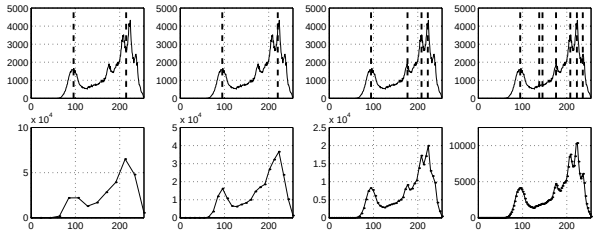

Fig. 4. Detection of histogram peaks using soft histograms.
Top: Detected peaks plotted in a 256-bin histogram.
Bottom: Soft histograms used.
13, 25, 50, *and* 100 *bins respectively.*

We can construct a soft histogram (with K bins) of the *double-angle local orientation* feature [7] $\boldsymbol{z}_n = m_n e^{\boldsymbol{i}\varphi_n}$ as follows:

$$h_k = \sum_{n=1}^{N} m_n \psi_k(\varphi_n) \quad \text{where} \tag{6}$$

$$\psi_k(\varphi_n) = \begin{cases} \cos^2\left(\frac{\omega}{d}(\varphi_n - \alpha_k)\right) & \text{if } \operatorname{adist}(\varphi_n - \alpha_k) < \frac{\pi d}{2\omega} \\ 0 & \text{otherwise} \end{cases}$$

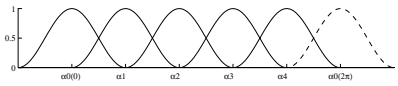

Fig. 5. Modular channels.

That is, we weight the contribution from each orientation value, φ_n, with the local orientation magnitude m_n. Since the angle φ_n only assumes values in the range $[0, 2\pi[$, we specify our bin distances as:

$$d = \frac{2\pi}{K} \quad \text{and our bin centres as} \quad \alpha_k = d(k-1)$$

Note that since φ_n is a modular variable, we have replaced the absolute distance in equation 2 with the angular distance. This implies that the first, and the last channels are neighbours (see figure 5).

If we have two such orientation histograms from two images that differ only by a rotation, we can compute this rotation from the phase of the discrete Fourier transforms of the two histograms, as described in [10]. Note that finding the correct rotation of the histogram, only means that we have found the image rotation modulo π, due to the double angle nature of local orientation. Full $360°$ rotation can however be found using soft histograms of the DIV operator described in [9].

7. EXPERIMENTS

As input feature to the soft histograms we will use local orientation vectors computed from the orientation tensors described in [8]. The tensor estimation uses 9 one-dimensional

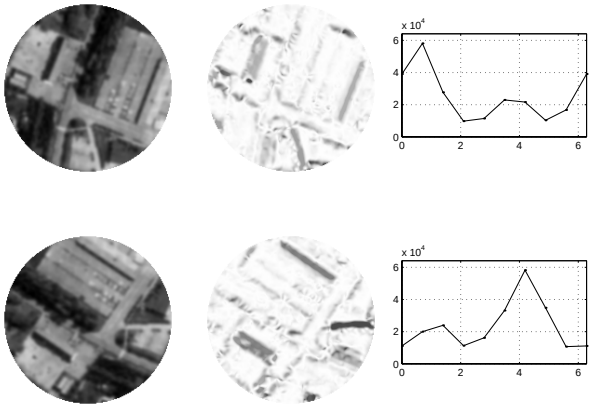

Fig. 6. Soft histograms of local orientation.
Left to right: Intensity images, Local orientation magnitudes $|\boldsymbol{z}|$, and soft histograms of two rotations of an image.

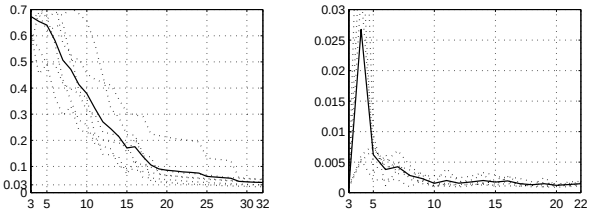

Fig. 7. Errors in estimation of rotation angle (radians).
Rotation estimation using conventional histograms (left) and soft histograms (right). Single image errors (dotted) and average (solid). Note the different scalings of the axes.

filters oriented along the horizontal and vertical axes, all with a spatial extent of 9 pixels.

From the orientation tensors, we compute a complex valued double angle image $\boldsymbol{z}(\mathbf{x})$ as follows:

$$\boldsymbol{z}(\mathbf{x}) = t_{11} - t_{22} + \boldsymbol{i}2t_{12}$$

Where t_{ij} are the components of the 2×2 tensor. This expression is equivalent to $(\lambda_1 - \lambda_2)\hat{\boldsymbol{z}}_1$ where $\lambda_1 \geq \lambda_2$ are the two eigenvalues, and $\hat{\boldsymbol{z}}_1 = e^{\boldsymbol{i}2\varphi}$ is a double angle representation of the principal subspace.

To evaluate the performance of the alignment procedure, we will now compute local orientation of an image, and the same image rotated varying amounts about the origin. We then use all orientation responses for which $x^2+y^2 = r^2 < 90$ to compute modular soft histograms according to equation 6 (see figure 6). We will use an overlap of $\omega = \pi/3$, and vary the number of soft bins K.

As a performance measure, we will use the absolute estimation error averaged over 45 different rotations in the range $(-\pi/2, \pi/2)$, and 6 different image regions. The result is compared with conventional histograms in the diagrams of figure 7. As can be seen, the accuracy is between 20 and 100 times better for the same number of bins.

An interesting observation is that the error is very small for soft histograms with just 3 bins. This is because the bins used span the entire range of the rotation angle, resulting in a histogram that only contains one single frequency component. For more realistic situations this kind of bins will not be useful, since they require an almost exact correspondence between the two images to match.

8. REFERENCES

[1] R. M. Gray, "Dithered quantizers," *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 805–812, 1993.

[2] C. M. Hicks, "The application of dither and noise-shaping to nyquist-rate digital audio: an introduction.," Technical report, Communications and Signal Processing Group, Cambridge University, UK, 1995.

[3] G. H. Granlund, "The complexity of vision," *Signal Processing*, vol. 74, no. 1, pp. 101–126, April 1999, Invited paper.

[4] K. Nordberg, G. Granlund, and H. Knutsson, "Representation and Learning of Invariance," in *Proceedings of IEEE International Conference on Image Processing*, Austin, Texas, November 1994, IEEE.

[5] M. Rosenblatt, "Remarks on some nonparametric estimates of a density function," *Annals of Mathematical Statistics*, vol. 27, pp. 642–669, 1956.

[6] E. Parzen, "On estimation of probability density function and mode," *Annals of Mathematical Statistics*, vol. 33, pp. 1065–1076, 1962.

[7] G. H. Granlund and H. Knutsson, *Signal Processing for Computer Vision*, Kluwer Academic Publishers, 1995, ISBN 0-7923-9530-1.

[8] G. Farnebäck, "Spatial Domain Methods for Orientation and Velocity Estimation," Lic. Thesis LiU-Tek-Lic-1999:13, Dept. EE, Linköping University, SE-581 83 Linköping, Sweden, March 1999, Thesis No. 755, ISBN 91-7219-441-3.

[9] Björn Johansson and Gösta Granlund, "Fast Selective Detection of Rotational Symmetries using Normalized Inhibition," in *Proc. of the 6th ECCV*, Dublin, Ireland, June 2000, vol. I, pp. 871–887.

[10] Per-Erik Forssén and Björn Johansson, "Fractal coding by means of local feature histograms," Report LiTH-ISY-R-2295, Dept. EE, Linköping University, SE-581 83 Linköping, Sweden, September 2000.