Chapter 4

Applications of Locally Orderless Images

In a recent work [141], Koenderink and Van Doorn consider a family of three intertwined scale-spaces coined the \textit{locally orderless image} (LOI). The LOI represents the image, observed at inner scale \( \sigma \), as a local histogram with bin-width \( \beta \), at each location, with a Gaussian-shape region of interest of extent \( \alpha \). LOIs form a natural and elegant extension of scale-space theory, show causal consistency and enable the smooth transition between pixels, histograms and isophotes. The aim of this work is to demonstrate the wide applicability and versatility of LOIs. We present applications for a range of image processing tasks, including new non-linear diffusion schemes, adaptive histogram equalization and variations, several methods for noise and scratch removal, texture rendering, and texture segmentation.

Histograms are ubiquitous in image processing. They embody the notion that for many tasks the distribution of intensity values within a region of interest contains the required information, with the spatial order of the pixels being immaterial. One can argue that even at a single location (which is a physical impossibility since any measurement is of non-zero scale) the intensity has an uncertainty, and should therefore be described by a probability distribution: physical plausibility requires non-zero imprecision. This led Griffin [89] to propose a \textit{scale-imprecision space} with spatial scale parameter \( \sigma \) and an intensity, or tonal scale \( \beta \), which can be identified as the familiar \textit{bin-width} of histograms.

Koenderink and Van Doorn [141] extended this concept to \textit{locally orderless images} (LOIs), an image representation with three scale parameters in which there is no local but only a global topology defined. LOIs are \textit{local histograms}, constructed according to scale-space principles, \textit{viz.} without violating the causality principle [136]. As such, one can apply to LOIs the whole machinery of techniques that has been developed in the context of scale-space research, such as the N-jet for local image structure description [137, 70, 94], the construction of invariants and (oriented) filter families [138, 139, 140, 71], non-linear diffusion schemes [195, 93], scale selection methods [160,
The aim of this chapter is to demonstrate that LOIs are a versatile and flexible framework for image processing applications; it may be viewed upon as a broad feasibility study. We do not give thorough evaluations for each of the applications presented. Obviously, local histograms are in common use in many areas, and the notion to consider histograms at different scales (soft binning) is not new either. Yet we believe that the use of a consistent mathematical framework in which all scale parameters are made explicit can aid in the design of effective algorithms by reusing existing scale-space concepts. Additional insight may be gained by taking into account the behavior of LOIs over scale.

**Locally orderless images**

We first briefly review locally orderless images \[141\] and comment on some interesting attributes. A convenient introduction is to consider the scale parameters involved in the calculation of a histogram:

- the inner scale \(\sigma\) with which the image is observed;
- the outer scale, or extent, or scope \(\alpha\) that parameterizes the size of the field of view over which the histogram is calculated;
- the scale at which the histogram is observed, tonal scale, or bin-width \(\beta\).

First of all, note that these scale parameters are distinct. Increasing \(\sigma\), i.e. blurring the image, has a different effect on the histogram than increasing \(\beta\), i.e. blurring the histogram itself. Consider a black-and-white image. The histogram contains two peaks and blurring the histogram will change this eventually to a uniform distribution. Blurring the image, on the other hand, will eventually create a uniform image, whose histogram is a delta peak.

The locally orderless images \(H(x, i; \sigma, \alpha, \beta)\) are defined as the family of histograms, i.e. a function of the intensity \(i\), with bin-width \(\beta\) of the image observed at scale \(\sigma\) calculated over a field of view centered around \(x\) with an extent \(\alpha\). The unique way to decrease resolution without creating spurious resolution is by convolution with Gaussian kernels [136, 261]. Therefore Gaussian kernels are used for both \(\sigma\), \(\alpha\) and \(\beta\).

We summarize this with a recipe for calculating LOIs:

1. Choose an inner scale \(\sigma\) and blur the image \(L(x; \sigma)\) using the diffusion

\[
\Delta_{(x)} L(x; \sigma) = \frac{\partial L(x; \sigma)}{\partial \sigma^2}.
\]  \hspace{1cm} (4.1)

with the boundary condition that \(L(x; 0)\) is the input image.
2. Choose a number of (equally spaced) bins of intensity levels $i$ and calculate the “soft isophote images”, representing the “stuff” in each bin through the Gaussian gray-scale transformation

$$R(x, i; \sigma, \beta) = \exp(- \frac{(L(x; \sigma) - i)^2}{2\beta^2})$$  \hspace{1cm} (4.2)

3. Choose a scope $\alpha$ for a Gaussian aperture, with unit amplitude

$$A(x; \alpha) = \exp(-\frac{x^2}{2\alpha^2})$$  \hspace{1cm} (4.3)

and compute the locally orderless image through convolution

$$H(x, i; \sigma, \alpha, \beta) = \frac{A(x; \alpha)}{2\pi\alpha^2} \otimes R(x, i; \sigma, \beta).$$  \hspace{1cm} (4.4)

Note that $R(x, i; \sigma, \beta)$ and $H(x, i; \sigma, \beta, \alpha)$, are stacks of isophote images, and therefore have a dimensionality 1 higher than that of the input image.

4. The bin-width $\beta$ can be increased by keeping $x$, $\sigma$, and $\alpha$ fixed and blurring along the intensity direction:

$$\frac{\partial^2 H(x, i; \sigma, \alpha, \beta)}{\partial i^2} = \frac{\partial H(x, i; \sigma, \alpha, \beta)}{\partial \frac{\beta^2}{2}},$$  \hspace{1cm} (4.5)

5. The extent $\alpha$ can be increased by keeping $i$, $\sigma$, and $\beta$ fixed and blurring each “intensity slice”:

$$\Delta_{(x)} H(x, i; \sigma, \alpha, \beta) = \frac{\partial H(x, i; \sigma, \alpha, \beta)}{\partial \frac{\alpha^2}{2}},$$  \hspace{1cm} (4.6)

For an illustration of the construction of an LOI, see Figure 5.1. The term *locally orderless* image refers to the fact that we have at our disposal at each location the probability distribution, which is a mere orderless set; the spatial structure within the field of view $\alpha$ centered at $x$ has been completely disregarded. This is the key point: with each spatial location we do not longer associate a (scalar) intensity, but a probability distribution, parameterized by $\sigma, \alpha, \beta$. Since a distribution contains more information then the intensity sec, we may expect to be able to use this information in various image processing tasks.

The LOI contains several conventional concepts, such as the original image and its scale-space $L(x; \sigma)$ that can be recovered from the LOI by integrating $iH(x, i; \sigma, \alpha, \beta)$
Figure 4.1: Examples of soft isophote images. The input image on the left has size 310 × 357 pixels and intensity values are distributed in the range [0,255]. The triple of numbers below each image indicates the intensity value $i$, inner scale $\sigma$ and tonal scale $\beta$, respectively. Note that, because $\beta > 0$ the isophotes are indeed soft: at some positions they fade gradually from white to black, at other positions this transition is very sharp.

over $i$. The image histogram in the conventional sense is obtained by letting $\alpha \to \infty$. The construction also includes families of isophote images, which for $\beta > 0$ are named soft isophote images by Koenderink (see Figure 4.1). And maybe even more important, by tuning the scale parameters the LOI can fill intermediate stages between the image, its histogram and its isophotes. This can be useful in practice. Finally, we note that the framework generalizes trivially to $n$D images or color images, if a color metric is selected.

Median and maximum mode evolution

If we replace the histogram at each location with its mean, we obtain the input image $L(x; \sigma)$ blurred with a kernel with width $\alpha$. This holds independently of $\beta$, since blurring a histogram does not alter its mean. If, however, we replace the histogram with its median or its maximum mode (the intensity with the highest probability), we obtain a diffusion with scale parameter $\alpha$ that is reminiscent of some non-linear diffusion schemes (see Figure 4.2). In fact it has been shown by Guichard and Morel \cite{91} that mean curvature flow corresponds to iterated median filtering, but the relation between other histogram operations and non-linear diffusion schemes is less clear.

The tonal scale $\beta$ works as a tuning parameter that determines the amount of non-linearity. Griffin \cite{89} proved that for infinite imprecision ($\beta \to \infty$) median filtering corresponds to linear filtering. With only a few soft isophote level images in the LOI, maximum mode diffusion also performs some sort of quantizing, and one obtains piecewise homogeneous patches with user-selectable values. This can be useful, e.g. in coding for data compression and knowledge-driven enhancements.
Figure 4.2: Input image is a sagittal $256 \times 256$ MR brain image with intensities in the range $[0,255]$. Each row shows a diffusion of the LOI with $\sigma = 0$ for $\alpha = 1, 2, 4, 8, 16$. Top row: mean diffusion, equivalent to linear scale space. Row 2 to 4: median diffusion for $\beta = 64, 16, 1$. Row 5 to 7: maximum mode diffusion for $\beta = 64, 16, 1$. 
Switching modes in bi-modal histograms

Instead of replacing each pixel with a feature of its local histogram, such as the median or the maximum mode, we can perform more sophisticated processing if we take the structure of the local histograms into account. If this histogram is bi-modal, this indicates the presence of multiple “objects” in the vicinity of that location.

Consider locations with bi-modal histograms (histograms with two maxima, with a single minimum in between). We let pixels at such locations “switch” to a desired mode. That is, if they are in the high mode (their value is higher than that of the minimum in between the two modes), we replace their value with the low mode (or vice versa, depending on the desired effect). The idea behind this is that a bright/dark object is replaced with the most likely value that the darker/brighter object in its vicinity has, namely the low/high maximum mode. Note that this a two-step process: the detection of bi-modal locations is a segmentation step, and replacing pixels fills in a value from its vicinity, thus using statistical information, and taking into account only those pixels that belong to the object to be filled in.

This scheme allows for a scale-selection procedure in the following way. For fixed $\sigma, \beta, \alpha$, there may be locations with more than two modes in their local distribution. This indicates that it is worthwhile to decrease $\alpha$, focusing on a smaller neighborhood, until just two modes remain. Thus we use a locally adaptive $\alpha$, ensuring that the replaced pixel value comes from information from locations “as close as possible” to the pixel to be replaced.

We have applied this scheme successfully for the removal of text on a complicated background (Figure 4.3), the detection of dense objects in chest radiographs, and noise removal. Figure 4.4 shows how shot noise can be detected and replaced with a probable value, obtained from the local histogram. The restoration is near perfect. We also applied the mode switching technique to old movie sequences with severe deteriorations. Most artifacts could be removed automatically.

Histogram transformations

Any generic histogram can be transformed into any other histogram by a nonlinear, monotonic gray-level transformation. To see this, consider an input histogram $h_1(i)$
Figure 4.4: Top-left: original image, 400 × 267 pixels, intensities scaled to [0, 1]. Top-right: original image with 20% shot noise. This is the input image for the restoration procedure. An LOI with $\sigma = 0, \beta = 0.04, \alpha = 0.4$ was computed. Middle-left: binary image with locations in top-right image with bi-modal histograms shown in white. Middle-right: binary image with locations in top-right image with bi-modal histograms and pixels in the low mode shown in white. These locations are replaced. Bottom-left: restoration using mode-switching for bi-modal locations gives excellent results. Bottom-right: restoration using a 5×5 median filter removes most but not all shot noise, and blurs the image.

and its cumulative histogram $H_1(i) = \int_{-\infty}^{i} h_1(i')$ and the desired output histogram $h_2(i)$ and $H_2(i)$. If we replace every $i$ with the $i'$ for which $H_1(i) = H_2(i')$ we have transformed the cumulative histogram $H_1$ into $H_2$ and thus also $h_1$ into $h_2$. Since cumulative histograms are monotonically non-decreasing, the mapping is monotonically
Figure 4.5: A normal PA chest radiograph of 512 by 512 pixels with intensities in the range [0,1]. Top-left: original image, in which details in lung regions and mediastinum are not well visible due to the large dynamic range of gray levels. Top-right: adaptive histogram equalization (AHE) based on the LOI with $\sigma = 0$, $\alpha = 8; \beta = 0.4$. Bottom-left: AHE with $\sigma = 0$, $\alpha = 8; \beta = 0.2$. Bottom-right: AHE with $\sigma = 0$, $\alpha = 4; \beta = 0.2$.

non-decreasing as well.

Although in practice this can only be achieved to limited accuracy due to discretization of the intensity domain, histogram manipulation is a very useful technique if one has an idea of what the output histogram as a function of the input histogram should be.

Histogram equalization is an example. The idea is that when displaying an image with a uniform histogram (within a certain range), all available gray levels or colors will be used in equal amounts and thus “perceptual contrast” is maximal. The idea to use local histograms (that is, selecting a proper $\alpha$ for the LOI) for equalization, to obtain optimal contrast over each region in the image stems from the 1970s [108] and is called adaptive histogram equalization (AHE). Several variations on this theme have been proposed for pre-processing of medical images, notably chest radiographs [35].
Figure 4.6: Top-left is an image (332 by 259 pixels) of rough concrete viewed frontally, illuminated from 22°. Top-right: the same material illuminated from 45°. Bottom-left shows the top-left image with its histogram mapped to the top-right image to approximate the change in texture. Bottom-right shows the result of local histogram transformation, with $\alpha = 2$. The approximation is especially improved in areas that show up white in the images on the left. These areas are often partly shadowed with illumination from 45°, and using a local histogram may correctly “predict” such transitions.

However, it was noted that these operations blow up noise in homogeneous regions. Pizer et al. [199] proposed to *clip* histograms, viz. for each bin with more pixels than a certain threshold, truncate the number of pixels and redistribute these uniformly over all other bins. It can be seen that this ad hoc technique amounts to the same effect as increasing $\beta$ in the LOI; notably, for $\beta \rightarrow \infty$, AHE has no effect. Thus we see that the two scale parameters $\alpha$ and $\beta$ determine the size of structures that are enhanced and the amount of enhancement, respectively. Figure 4.5 shows a practical example of such a continuously tuned AHE for a medical modality (thorax X-ray) with a wide latitude of intensities.

An alternative to histogram equalization is to increase the standard deviation of the histogram by a constant factor, which can be done by a linear gray level transformation. Again, the LOI provides us with an elegant framework in which the
Figure 4.7: (a) A texture from the Brodatz set [26], resolution $256^2$, intensities in the range $[0, 1]$. (b) Blurred Gaussian noise, scaled to range from $[0, 1]$. (c) Multiplication of (a) and (b). (d) Reconstruction of (a) from (c) from the LOI with $\sigma = 0, \beta = 0.1, \alpha = 2$ and computing for each point a mapping to the local histogram at the randomly chosen location (80,80).

scale parameters that determine the results of such operations are made explicit.

Another application of histogram transformation is to approximate changes in texture due to different viewing and illumination directions [81]. In general, the textural appearance of many common real-world materials is a complex function of the light field and viewing position. In computer graphics it is common practice, however, to only apply a projective transformation to a texture patch in order to account for a change in viewing direction and to adjust the mean brightness using a bi-directional reflection distribution function (BRDF), often assumed to be simply Lambertian. In [81] it is shown that this gives poor results for many materials, and that histogram transformations often produce far more realistic results. A logical next step is to consider local histogram transformations. An example is shown in Figure 4.6, using a texture of rough concrete taken from the CURET database [53]. Instead of using one mapping function for all pixel intensities, the mapping is now based on the pixel intensity and the intensities in its surroundings. Simple physical considerations make clear that this approach does make sense: bright pixels which have dark pixels due to shadowing in their neighborhood are more likely to become shadowed for more oblique illumination than those that are in the center of a bright region.

Finally, histogram transformations can be applied to restore images that have been corrupted by some noise process, but for which the local histogram properties are known or can be estimated from the corrupted image. Although this may sound contrived, we believe such cases are encountered frequently in practice. Many image acquisition systems contain artifacts that are hard to correct with calibration schemes. One example in medical image processing is the inhomogeneity of the magnetic field of an MR scanner or of the sensitivity of MR surface coils, leading to low frequency gradients over the image, for which retrospective correction schemes have been proposed [156,157]. A generated example is shown in Figure 4.7 where we multiplied a texture image with Gaussian noise. We make two assumptions: 1) the noise does not contain high frequency components and thus the structure of the local histogram will...
not be affected for a small scope $\alpha$; 2) the original texture image has small textons, so that over a small scope, the histograms are equal everywhere. By randomly choosing a point in the corrupted image and computing the mapping that transforms each local histogram to the local histogram at that particular location we obtain the restored image in Figure 4.7(d). The texture of the restored image is more uniform than in the original, indicating that assumption 2) is not completely valid. Furthermore, in areas where the noise factor with which the texture was multiplied was close to 0, all information is destroyed, and restoration fails. Note that we did not make any assumption about the noise process. In this particular case, dividing Figure 4.7 with a median filtered version of it, would give good restoration as well, but this assumes corruption by multiplicative noise. The LOI method works for additive noise, or other kinds of noise as well.

Texture segmentation based on local histograms

Many “general” (semi-)automatic segmentation schemes are based on the notion that points in spatial proximity with similar intensity values are likely to belong to the same object. Such methods run into problems with textured areas, because the intensity values may show wild local variations. A solution is to locally compute texture features and replace pixel values with these features, assuming that pixels that belong to the same texture region will now have a similar value. The framework of LOIs is ideally suited for the computation of such local features. The idea of using texture information sampled from a local aperture is not new. Shi and Malik [220] have applied their normalized cut segmentation scheme to texture segmentation in this way, using local histograms and the correlation between them as a metric. Gårding and Lindeberg [76] used an integration scale, comparable to $\alpha$, to denote the aperture.
Here we present an adapted version of a semi-automatic segmentation technique called seeded region growing (SRG), developed by Adams and Bischof [2], that is popular in medical image processing. An attractive property of our adaptation of this scheme is that it reduces to a scheme very similar to the original SRG scheme for $\alpha \to 0$. This is directly due to the fact that LOIs contain the original image.

SRG segments an image starting from (user-selected or in some way automatically determined) seeds. The algorithm maintains a list of all pixels that are connected to one of the seeds, sorted according to the distance of the pixel to the seed. This metric is originally defined as the intensity of the pixel minus the mean intensity of the seed region squared. The pixel at the top of the list is added to the seed-region it is connected to, the mean intensity of this region is updated and the neighbors of the added pixel are added to the list. This procedure is repeated, usually until all pixels are assigned to a region. The effect is that seeds in a homogeneous region grow quickly; once an edge is reached, the growing process grinds to a halt.

Instead of comparing pixel intensity values with the mean intensity of a region, we compare the local histograms of a pixel and a region. This requires the notion of a distance between two distributions and there are several ways in which this can be implemented. We choose to subtract the histograms and take the sum of the absolute values of what is left in the bins. For $\alpha \to 0$ this reduces to a scheme similar to the original scheme, except that one considers for the region the global mode instead of the mean of the histogram (most likely pixel value instead of the mean pixel value). Figures 4.8 to 4.10 illustrate the use of seeded region growing based on local histograms.

**Figure 4.9:** Left: a test image composed of 6 texture patches of pixel size $128 \times 128$ each. Intensity values in each patch are normalized to zero mean and unit variance. Right: the result of segmentation with seeded region growing based on an LOI with $\sigma = 0$, $\beta = 0.2$ and $\alpha = 8$. The circles are the seeds.
Applications of Locally Orderless Images

Figure 4.10: Top row, left: wildlife scene with leopard, size $329 \times 253$ pixels, intensities scaled between $[0,1]$; Bottom row, left: a locally ($\sigma = 8$) normalized version of the input image; Middle and right: segmentation by SRG based upon an LOI with $\sigma = 0$, $\beta = 0.05$ and $\alpha = 0.4$, respectively. Note how well the textured area is segmented in the lower right image.

Concluding remarks

In the applications presented, we have used many aspects of LOIs. In some cases, they are a natural extension ($\alpha > 0$) of techniques that usually use pixels ($\alpha = 0$), e.g. seeded region growing. In other cases, they extend techniques that use “conventional” histograms ($\alpha \rightarrow \infty$) with an extra parameter, e.g. histogram transformation techniques. Other applications exploit the behavior of LOIs over scale to obtain non-linear diffusions and to remove noise. We conclude that LOIs are image representations of great practical value.