Chapter 4

Multi-scale Analysis

4.1 Introduction

At its core, image processing is simply a computer manipulation of a matrix of luminosity values. The manipulation may be complex, for example the reconstruction of a full image using only a small number of pixels from the original image, or quite simple, for example reducing the resolution or size of an image. Due to the development of digital technology and the vast amount of image data readily available there is an increasing need for computer based image processing and analysis. It may be a manipulation to improve a photograph for viewing purposes, such as deblurring, sharpening, colour manipulation or red eye reduction. These are examples of our most common requirements as humans from image editing. The relatively recently developed field of computer vision has seen the addition of other requirements. This field involves developing methods which allow a computer or robot to analyze an image automatically, similar to the way in which the human vision system (HVS), that is the eyes plus the brain or visual cortex, analyzes its surroundings. With the computer representing the brain, the camera the eyes, and the video or image captured the surroundings, the ultimate aim is to eliminate human involvement in the process. For example, consider applications in security such as target detection, identification and tracking, applications in medical imaging to automatically detect anomalies thereby enforcing a doctor's findings, or applications in industry to detect when a production plant is producing reduced quality products.

Obviously the human vision system is highly complicated. The introductions

of [252] and [84] provide insight into what little we do know about the brain and the visual cortex. Research into the brain has provided only glimpses of how subprocesses work and how we could relate the brain to our minds or consciousness. The question of the fruitfulness of computer vision then arises. If we are attempting to model a human action, but we do not understand how it works, how can we hope to replicate this action? It is obvious that the human brain learns as it progresses through life thus offering the solution to a complicated process - we need to provide our techniques with useful extracted information and then teach our algorithms to learn from this data. It is human nature to be inquisitive and research in computer vision can only advance. At some point brain research will have reached heights we cannot now fathom, as with scientific research over the centuries. It only makes sense then to attempt to keep up in the field of computer vision, and to model our computer visual system as closely as possible to the HVS for now, and as advances are made in brain research this can replicated in computer vision. This said, methods which do not attempt to model the HVS should not be discredited at all. The possibility of a smarter method is worth the departure from modeling reality and a computer is obviously not a brain so we cannot hope to treat them in the same manner. The field of super resolution imaging is such an example. It provides the creation of a scene at a higher resolution using a number of lower resolution captures of the same scene.

This chapter deals with changing the representation of an image from two to three dimensions by adding a scale parameter. The aim of this is to provide a more efficient representation in the sense that certain aspects of the information contained in the image is immediately accessible [129]. The original two-dimensional matrix representation of an image is implicit in the sense that the information is contained therein but is not directly accessible. The idea of adding a scale parameter stems from the observation that objects in our surroundings occur at different scales, either due to their size or their resolution with respect to the observer. All around us we view objects of different sizes and this is transferred into an image when a scene is captured on camera. Thus an image is made up of objects of varying sizes, or specifically varying scales. The content of an image can each be present at more than one scale with each scale representing information of varying importance or detail. Consider an image of a face brick house. The wall can be identified as a relatively large flat structuring element consisting of small rectangular elements each porous in texture and varying shading, that is, the presence of at least three different scales can be seen. This illustrates the importance of being able to extract information at various scales. Thus the natural multi-

scale nature of our surroundings encourages a model in which we include the scale as a parameter. To connect the concepts of scale and the DPT pulses, we consider the number of pixels included in a connected region , that is the area of the connected region, [213] as its scale.

The next obvious question is which scale(s) to focus on. A specific object will only be present in a certain range of scales, but this range could differ for another object or even for the same object is a different scene. In addition, as described in [109] there exists 'outer' and 'inner' scale restrictions in any image we analyze, which refer to the maximum extent of the image window or frame, that is, the coarsest detail, and the maximum resolution of the image, that is, the smallest detail which can be observed, respectively. These scale restrictions will differ from image to image as well, making matters more complicated. For example, an object in a lower resolution as well as smaller image will be present in a different range of scales to the same object in an image with higher resolution and larger size, since the base scale, namely 1, is measured at the individual pixel level. The 'inner' scale also poses an additional restriction compared to traditional numerical methods. The maximum resolution is restricted so that we cannot increase it further to improve our results as we do not have data at a higher resolution, as one can do in approximation theory by making the approximation points gradually closer and closer together [123]. There are also numerous operators available which we can apply to images. Thus one may ask, what operator should I use?, where should I use it exactly? and, what size scale should it act on? [129]. This all depends on where the meaningful information in the image. As discussed, this differs from image to image and also depends on the interest of the observer. The nature of the problem at hand is also ill-posed [123]. Recall that according to Hadamard [74, 174] a problem is well-posed if a solution exists, is unique and the solution depends continuously on the data. The projection on a two dimensional image of any three dimensional object or scene, except for simple cases such as a solid smooth sphere, may result in an infinite number of different possible shapes.

The above thus presents a strong case for a representation which treats *all* scales equally at first without any a priori information about the scene in the image. This is in alignment with the universal physical law of scale invariance, the Pi-theorem, namely that physical laws must be independent of the choice of fundamental parameters or in other words a function relating physical observables must be independent of the choice of dimensional units, i.e. no change over scale [64]. Thus we assume nothing for our computer visual front-end and initially consider all possible scales. This also allows

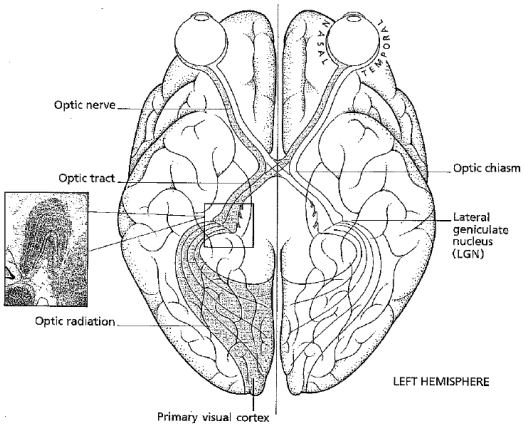


Figure 4.1: *Path of impulses received by the retina to the visual cortex (from [252, figure 3.3, page 23])*

for a large degree of generality in order for the model to be able to solve a large number of problems [123]. Making use of scale-spaces or multiscale methods to analyze an image allows the use of more information than the pixel luminosity only. This allows for providing our algorithms with all the necessary information with which to learn, and not restricting the view to those scales we a priori assume are important. A scale-space is formally (and in its original Gaussian form) the representation of an image f as a continuous family $\{T_t(f) : t \geq 0\}$ of gradually smoother versions of it [238]. The original image is represented at $t = 0$. As described by [168] it is 'an ordered stack of pictures each representing the same scene but at a different level of detail'. This representation provides us with more in-depth information than when the image is represented in its original form so this calls for an effective way to manage this new data and an effective way in which to reduce it to the significant information it is providing us with [246].

How does the human visual system operate? It is interesting to note that the HVS is the best understood part of our complicated brains [84]. As seen in Figure 4.1 the optic nerve carries the impulses received from the retina in the eyes across the optic chiasm to the visual cortex. Note the interesting routes the impulses travel.

There is strong evidence for a 'perceptive' cortex, the striate cortex, and an 'association' cortex, the cortex surrounding the striate cortex [252, Chapter 6]. This is interpreted as the HVS consisting of two stages [105, 156]: a pre-attentive stage in which 'pop-out' [104] features are detected, and then an attentive stage in which relationships between the features are detected and

grouping takes place. The 'pop-out' features are considered salient, that is more discriminating in some way. This idea of two-stage vision also influenced Marr in his book *Vision* [138].

In addition there is strong evidence for a hierarchical process in the HVS [252, Chapter 9]. An image of the surroundings is not projected into our eyes and transmitted as is to the visual cortex. Experiments by Hubel and Wiesel have proven the existence of a process at work which analyzes what we see in a hierarchical or layered manner, that is, takes in the surroundings as a number of building blocks which then make up the entire scene. Interestingly there is also evidence of the HVS working in a parallel manner [252, Chapters 11, 13] with multiple visual areas processing separate things individually.

So our argument for the use of scale-spaces, a representation of an image at all its scales, is strong if we prefer to make no prior assumptions about the scene and allow the algorithm every possible piece of information to work with. We now give a short overview of the scale-spaces researched before the most famous linear scale-space of Witkin was introduced.

The pyramid was the first approach to strictly treating an image in a hierarchical manner, but some pioneers did investigate looking at multiple scales at a time [129], namely, Rosenfeld and Thurston in 1971 who used operators of different sizes for edge detection [191], and [108, 226, 76, 224] who investigated sub-sampling by different amounts.

The basic idea of a pyramid involves the concept of a quadtree. This is obtained via recursive decomposition [202, 129, 123]. More specifically it is the successive subdivision of an image into four equally sized quadrants until the blocks obtained at some subdivision are homogeneous, for example, consist only of 1's and 0's in a binary image. For a greyscale image a measure Σ is defined to measure the homogeneity of the quadrants. It could be for example standard deviation or a thresholding between maximum and minimum pixel luminosity values. Consider an image f of size $2^K \times 2^K, K \in \mathbb{Z}$ and some subdivision $f^{(k)}$. If $H(f^{(k)})$, the measure of homogeneity of $f^{(k)}$, is too large according to some specific value, $f^{(k)}$ is split into $f_j^{(k-1)}, j = 1, 2, ..., p$ according to some rule. Generally p is taken as 4, thus referring to the resulting tree as a quadtree where the leaves $f_j^{(k-1)}$ are homogeneous. This is applied recursively to each subimage $f_j^{(k-1)}$ until the homogeneity of each subimage is satisfied.

This method can in fact be viewed as a simple segmentation algorithm and has been adapted into the 'split-and-merge' algorithm in which adjacent ho-

mogeneous regions (or quadrants) are merged if their measures of homogeneity are similar.

The pyramid is a version of the quadtree but includes a smoothing step at each subdivision as well, see [202, 129, 128, 123], and can be credited to Burt [31] and Crowley [43] individually. The subdivision is in fact a size reduction step so that the image size decreases exponentially with the scale level. The main advantage of a pyramid is that the reduction in image size leads to reduced computational work. For example, consider a low-pass pyramid of Burt and Crowley for a discrete one-dimensional signal f,

$$f^{(k-1)}(x) = \sum_{n=-N}^{N} c(n) f^{(k)}(2x-n)$$

with filter coefficients $c(n), n = -N, ..., N$. Criteria with respect to the coefficients include positivity $c(n) \geq 0$, unimodality $c(|n|) \geq c(|n+1|)$, symmetry $c(-n) = c(n)$, normalization $\sum_{n=-N}^{N} c(n) = 1$, and equal contribution. The *equal contribution* criterion ensures that all pixels contribute equal amounts to all levels by requiring the sum of the weights remains constant over the levels.

In choosing coefficients [143] proposed that an ideal low-pass filter should be approximated as best as possible. A low-pass pyramid involves a smoothing filter first and then a subsampling of the image at each step. Examples include the Gaussian pyramid and Laplacian pyramid, the latter of which is a bandpass pyramid obtained as the difference between two adjacent levels of a low-pass pyramid like the Gaussian pyramid. These have been used in feature detection and image compression.

Wavelets are another early example of incorporating scale into the analysis. The wavelet transform [133, 45, 147] was developed as an improvement over the window Fourier transform.

The continuous wavelet transform [71, 229, 132, 45, 180] decomposes a signal over a set of translated and dilated versions of a 'mother wavelet' $\psi \in \mathcal{L}^2(\mathbb{R})$ which has zero mean $\int_{\mathbb{R}} \psi(t)dt = 0$, is normalized $\|\psi\|_{\mathcal{L}^2} = 1$ and is centered at 0. The fact that ψ has zero mean also implies that the function must be oscillatory and therefore is a wave. For various dilation and translation parameters a and b a set of wavelets

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi \left(\frac{t-b}{a} \right)$$

is obtained. The simplest example of a wavelet is the Haar wavelet, see [45, 52]. The continuous wavelet transform is then a function of the two new parameters a and b,

$$CWTf(a,b) = \langle f, \psi_{a,b} \rangle_{\mathcal{L}^2} = \int_{\mathbb{R}} f(t) \frac{1}{\sqrt{a}} \psi^* \left(\frac{t-b}{a} \right) dt$$

where ψ^* indicates the complex conjugate of ψ, and it decomposes f with respect to wavelet basis set. The function can be fully recovered via the inverse wavelet transform

$$f(t) = \int \int CWTf(a,b) \psi_{a,b}(t) da db.$$

If $\psi(t)$ satisfies the following admissibility criterion,

$$\int \frac{|\Psi(\omega)|^2}{|\omega|} d\omega < \infty$$

where Ψ is the Fourier transform of ψ, then ψ can be used to analyze and reconstruct the signal without loss of information [229]. Additional regularity conditions are also imposed by [147], namely that the wavelet transform decreases quickly with scale. The family of wavelets is considered redundant [45] thus an orthogonal basis of wavelets is preferred [133, 132, 45]. In higher dimensions the wavelet transform is simply the combination on a product space of a number of separable one dimensional transforms [78].

In order to apply the wavelet transform to digital signals a discrete theory is needed. This is simply attained by discretizing the continuous wavelet transform. The set of wavelets become

$$\psi_{j,n}(m) = \frac{1}{\sqrt{s_0^j}} \psi(s_0^{-j} m - n)$$

where ψ is the original continuous mother wavelet, $k \in \mathbb{Z}$, and s_0^j indicates a dilation of resolution s_0^j ($s_0 = 2$ corresponds to dyadic sampling). The signal is then also discretized by sampling it at points $m = 1, ..., N$. The discrete wavelet transform is then

$$DWTf(n,j) = \sum_m f(m) \psi_{j,n}^*(m).$$

The discrete signal can similarly be fully recovered here,

$$f(m) = \sum_{n,j} DWTf(n,j) \psi_{j,n}(m).$$

The wavelet transform is useful for compression by efficiently and effectively sampling from the parameters a and b [180]. In [180] a discrete time wavelet theory is developed by redefining what is meant by discrete scale and resolution through the sampling rate - they do not simply discretize the continuous theory.

4.2 Background Theory

For simplicity we present some background theory here which is needed later in this chapter.

The Gaussian

The univariate normal distribution [101] is given by

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\left((x-\mu)/\sigma\right)^2/2}$$

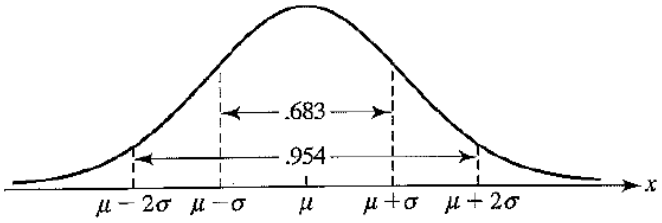
for $x \in \mathbb{R}$. More precisely, if a random variable X has a density function $f(x)$ as given above we say that X is distributed normally with mean μ and variance σ^2 (standard deviation σ), and we write $X \sim N(\mu, \sigma^2)$. The term $(x-\mu)^2/\sigma^2$ in the exponential exponent measures the distance from x to μ in standard deviation units.

The normal distribution in p dimensions [101], for a vector $\underline{X} = [X_1, X_2, ..., X_p]$, has the following form,

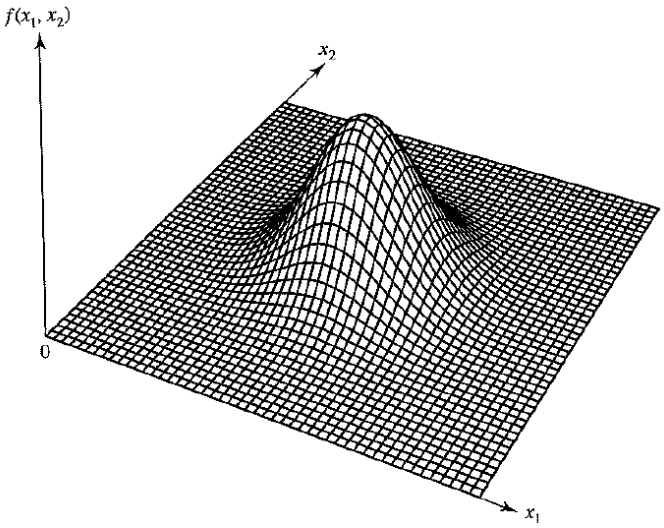
$$f(\underline{x}) = \frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}(\underline{x}-\underline{\mu})^T\Sigma^{-1}(\underline{x}-\underline{\mu})} \text{ for } -\infty < x_i < \infty, i = 1, 2, ..., p.$$

The vector random variable is then normally distributed with mean $\underline{\mu} = [\mu_1, \mu_2, ..., \mu_p]$ and covariance matrix Σ where the covariance matrix is required to be positive definite. We write $\underline{X} \sim N(\underline{\mu}, \Sigma)$. The term $(\underline{x} - \underline{\mu})'\Sigma^{-2}(\underline{x} - \underline{\mu})$ in the density function above is called the Mahalanobis distance and measures the square distance from the vector $\underline{x}$ to the mean $\underline{\mu}$ in the units of the covariances.

For the bivariate case $p = 2$ (for application in images) we consider X_1 and X_2 uncorrelated and with equal variances and means so that the correlation



(a)



(b)

Figure 4.2: *(a) One dimensional normal density function [101] with the areas under the curve indicated by the vertical lines (b) Two dimensional normal density function [101] which may be symmetric for the case of equal variances or spherical for the case of unequal variances*

matrix Σ has the form

$$\Sigma = \begin{bmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{bmatrix}.$$

Of course the uncorrelated case is also a logical choice, as well as the case of unequal variances.

Some properties of the Gaussian distribution are that any linear combination of the components of $\underline{X}$ is normally distributed, any subset of the components of $\underline{X}$ have a normal distribution and the conditional distributions of the components are normally distributed [101]. Figure 4.2 provides an illustration of the one and two dimensional normal densities.

Convolutions

A convolution [18] is an integral expression, involving two functions f and g, for the amount of correlation of g with f as g is shifted and flipped over f. In other words it blends the one function into the other. A beautiful moving illustration of the concept in shown on the webpage [240]. The convolution is defined as follows.

$$G(t) = (f * g)(t) = \int_0^t f(\tau)g(t-\tau)d\tau \text{ (for a finite range)} \quad (4.1)$$

$$= \int_{-\infty}^{\infty} f(\tau)g(t-\tau)d\tau = \int_{-\infty}^{\infty} g(\tau)f(t-\tau)d\tau. \quad (4.2)$$

The following properties hold for a convolution of f and g.

- $f * g = g * f$ (commutativity)
- $f * (g * h) = (f * g) * h$ (associativity)
- $f * (g + h) = (f * g) + (f * h)$ (distributivity)
- $a(f * g) = (af) * g = f * (ag)$ for a constant a
- $(f * g)' = f' * g = f * g'$ where $'$ is the derivative
- $F(f * g) = F(f) * F(g)$ where F is the Fourier transform (Convolution Theorem) [166]

Kernels

Schölkopf and Smola [205] provide an excellent work on kernels in computer learning. The first use of the kernel arose as a function in the field of integral operators [80, 39, 144]. A function k giving rise to an operator T_k via $(T_k(f))(x) = \int_{\mathcal{X}} k(x, x')f(x')dx'$ is called the *kernel* of T_k. More specifically a kernel k is a dot product of a feature space $\mathcal{H}$ via a mapping $\Phi : \mathcal{X} \to \mathcal{H}$, that is $k(x, x') = \langle \Phi(x), \Phi(x') \rangle$. The standard requirement on a kernel is positive definiteness. When $\mathcal{H}$ is $\mathbb{R}$ or $\mathbb{C}$ the kernel k is positive definite if the kernel matrix (Gram matrix) $K = [K_{ij}] = [k(x_i, x_j)]$ is positive definite, that is $\sum_{i,j} c_i \bar{c}_j k_{ij} \geq 0 \ \forall c_i \in \mathcal{H}$.

Name	Formulation
Homogeneous Polynomial Kernel	$k(x, x') = \langle x, x' \rangle^d$
Gaussian Kernel	$k(x, x') = \exp^{-\frac{1}{2\sigma^2}\|x-x'\|^2}$
Inhomogeneous Polynomial Kernel	$k(x, x') = (\langle x, x' \rangle + c)^d$ for $d \in \mathbb{N}, c \geq 0$
Radial Basis Function Kernel	$k(x, x') = f(d(x, x'))$ where d is a metric on $\mathcal{X}$, and f a function on $\mathbb{R}_0^+$

Table 4.1: Examples of Positive Definite Kernels

Some positive definite kernels are presented in Table 4.2. Other kernels include the cosine, Hilbert, exponential, B_n spline, rational quadratic, Bartlett, Daniell and Parzen kernels. There also exist kernels which are not symmetric [116]

In addition, if the solution of a partial differential equation, namely f, can be written as $T_k(f)$ above, then the kernel becomes the Green's function. For the heat or diffusion equation, the kernel is the Green's function. The heat kernel in $\mathbb{R}^d$ is as follows,

$$k_t(x, y) = \frac{1}{(4\pi t)^{d/2}} e^{-(x-y)^T(x-y)/4t} \; \forall x \in \mathbb{R}^d \text{ and for any } y \in \mathbb{R}^d.$$

The reader will notice the heat kernel is in fact the Gaussian. The heat kernel represents the evolution of temperature in a region whose boundary is held fixed at a particular temperature and a initial heat source is placed at a point at time 0 [16].

Kernel methods in machine learning include kernel principal components for feature extraction, kernel Fisher discriminant for feature extraction and classification, and Bayesian kernel methods to name but a few.

Modified Bessel Functions of Integer Order

The derivation of the discrete Gaussian scale-space involves in the use of the modified Bessel function of integer order. We present them here for simplicity.

The solutions of the differential equation

$$x^2\frac{d^2y}{dx^2} + x\frac{dy}{dx} + (x^2 + n^2)y = 0$$

are called the Bessel functions of order n [10, 237]. These were first defined by Daniel Bernoulli but then generalized by Friedrich Bessel. The two linearly dependent solutions, for an integer n, are given by $J_n(x)$ and $J_{-n}(x)$ where

$$J_n(x) = \left(\frac{1}{2}x\right)^n \sum_{k=0}^{\infty} \frac{\left(-\frac{1}{4}x^2\right)^k}{k!\Gamma(n+k+1)}.$$

The function $Y_n(x)$ is defined as

$$Y_n(x) = \frac{J_n(x)\cos(n\pi) - J_{-n}(x)}{\sin(n\pi)}$$

so that $J_n(x)$ and $Y_n(x)$ are linearly independent, and is called the Bessel function of the second kind (also known as Weber's function and Neumann functions).

The modified Bessel functions of integer order, $I_n(z)$, are obtained when allowing x to be complex but the result real. The solutions of the differential equation are then $I_n(z)$ and $I_{-n}(z)$ when n is not an integer and $I_n(z)$ and $K_n(z) = \frac{1}{2}\pi\frac{I_{-n}(z)-I_n(z)}{\sin(n\pi)}$ when n is an integer. In terms of the original Bessel functions,

$$I_n(z) = i^{-n}J_n(iz) = \sum_{m=0}^{\infty} \frac{\left(\frac{1}{2}x\right)^{n+2m}}{m!\Gamma(n+m+1)} = \frac{1}{\pi}\int_0^{\pi} \exp\{x\cos\tau\}\cos(n\tau)d\tau.$$

For more details, properties and results see [237, 178, 142, 159, 10].

4.3 Scale-Space History

In 1983 Witkin published the first work on the Gaussian scale-space [246, 245]. There was also a technical report from MIT by Stanfield in 1980 [220] which describes a first thought on scale-spaces, as mentioned in [239]. In 1984 Koenderink published an equivalent formulation to Witkin's as the solution of the linear diffusion process [109]. These are considered the foremost work on the linear Gaussian scale-space, which has now grown into a very well

known topic in computer vision. Further pioneering work has been done by Lindeberg, Weickert, Koendrink, ter Haar Romeny, Florack and Viergever, to name the most prominent.

However, contrary to the timeline above, it seems that scale-spaces were independently developed in Japan by Iijima in 1959, [97] - [88]. The work remained undiscovered by the western world, probably because the majority of the works were in Japanese, until 1997 when the connection between the two independent developments was provided in [239, 238]. Weickert also describes in these works that perhaps the research field of scale-spaces wasn't developed enough, or its importance thought of, in 1959 for Iijima's work to be appreciated and thus his work flew under the radar of computer vision scientists. In addition there are three other Japanese linear scale-space approaches that were developed before 1983. All the work presented below by Japanese scientists is from [239] and [238] where it is comprehensively summarized.

The oldest is Taizo Iijima's work done from 1959 [97, 86, 98, 87, 88]. Iijima was working, at the time, at the Electrotechnical Laboratory on optical character recognition and realized the need for a general framework for extraction of characteristic information from patterns. This first work was developed for one dimensional signals from simplicity and relies on four axioms, namely, linearity, translation invariance, scale invariance and a semigroup property. Iijima chose these axioms to remain in line with requirements for object recognition, that is, it should be invariant under changes in the reflected light intensity, parallel shifts in position, and expansions or contractions of the object. He also assumes that the observation results in a blurry transformation Φ and calls this class of blurring transformation 'BOKE' (defocusing). More specifically, with an original image $g(x)$ the blurred version obtained via a convolution with a kernel ϕ has the structure

$$\Phi(g,\sigma)(x) = \int_{-\infty}^{\infty} \phi_\sigma(x,x')g(x')dx' \tag{4.3}$$

where σ is an observation parameter. Four axioms are assumed to be satisfied by the transformation 4.3. They are *linearity w.r.t scalar multiplication* (if the image intensity becomes a times more, then the transformed intensity is similarly a times more), *translation invariance*, *scale invariance* and a generalized *semigroup property* (if g is observed at scale σ_1, and this is in turn observed at scale σ_2, then the equivalent observation scale is $\sigma_3(\sigma_1,\sigma_2)$ for some σ_3) [97, 86, 98, 87, 88, 239]. Note that for uniqueness of the scale-space Iijima claims preservation of positivity is needed as an additional axiom,

namely

$$\Phi\big(g, x, \sigma\big) > 0 \; \forall \; g(x) > 0, \; \forall \; \sigma > 0.$$

However, in his 2002 PhD thesis Felsberg [58] shows that the Poisson kernel also satisfies Iijima's 5 axioms of linearity, scale and shift invariance, a semi-group property and positivity preservation thus disputing the uniqueness under these specific axioms.

Iijima derives the following from (4.3)

$$\Phi\big(g, x, \sigma\big) = \frac{1}{2\sqrt{\pi}\sigma} \int_{-\infty}^{\infty} g(x') \exp\left\{-\frac{(x-x')^2}{4\sigma^2}\right\} dx',$$

which is a convolution between g and a Gaussian with standard deviation $\sigma\sqrt{2}$. Iijima also argues for Gaussian blurring as our visual perception is carried out through a lens which has a Gaussian-like blurring profile [88].

Iijima next generalized this derivation to two dimensions [98, 87]. His blurring transformation is as follows,

$$\Phi\big(f, x, \Sigma\big) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi(f(x'), x, x', \Sigma) dx'_1 dx'_2,$$

where Σ is a 2×2 symmetric positive definite matrix, $x' = (x'_1, x'_2), x = (x_1, x_2)$, and the four axioms are, similar to the one-dimensional case, linearity w.r.t multiplications, translation invariance, scale invariance and closedness under affine transformations, and a generalized semigroup property [98, 87].

If, in addition, positivity preservation is assumed then the blurring is called the affine Gaussian scale-space:

$$\begin{aligned}
\Phi\big(f, x, \Sigma\big) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x'_1, x'_2) \phi(x_1 - x'_1, x_2 - x'_2, \Sigma) dx'_1 dx'_2 \\
\text{with } \phi(u_1, u_2, \Sigma) &= \frac{1}{4\pi\sigma^2} \exp\left(- \frac{\mu_{22} u_1^2 - 2\mu_{12} u_1 u_2 + \mu_{11} u_2^2}{4\sigma^2} \right) \\
\text{and } \Sigma &= \sigma^2 \begin{bmatrix} \mu_{11} & \mu_{12} \\ \mu_{12} & \mu_{22} \end{bmatrix}, \det \begin{pmatrix} \mu_{11} & \mu_{12} \\ \mu_{12} & \mu_{22} \end{pmatrix} = 1.
\end{aligned}$$

To obtain the modern (Witkin and later) isotropic Gaussian scale-space kernel an axiom of invariance under rotations is further needed. This provides

a scale-space which is invariant with respect to direction and results in the third axiom of scale invariance and closedness under affine transformations being equivalent to ordinary pure scale invariance.

Iijima then further re-derived his scale-space in 1971 in order to obtain a more physically consistent formulation [239]. This work of his appears in [89, 91, 92, 90, 93, 94, 95, 96]. His idea is to generalize the original figure (signal or image) $f(r)$ to $f(r,\tau)$ such that the method attempts to model the defocusing of the HVS or an optical system. He assumes two principles, namely the *conservation principle* and the *principle of maximum loss of figure compression*. The conservation principle requires the transformation not to change the total energy of the image function so that the image function satisfies the continuity equation

$$\frac{\partial f(r,\tau)}{\partial \tau} + \nabla \cdot I(r,\tau) = 0$$

where I is the flux (flow per unit) for the figure flow, r is the location, τ the blurring parameter, and ∇ indicates divergence operation in $\mathbb{R}^2$. The continuity equation states that the rate at which the image function energy decreases is proportional to the outward flux. His second principle involves maximizing the figure flow, that is, maximizing

$$J(I) = \frac{|I^T \nabla f|^2}{I^T R^{-1} I}$$

where $R(\tau)$ is a positive definite matrix denoting the medium constant of the blurring process. This is maximized for $I(r,\tau) = -R(\tau) \cdot \nabla f(r,\tau)$. These two principles result in the anisotropic linear diffusion equation

$$\frac{\partial f(r,\tau)}{\partial \tau} = \nabla \cdot (R(\tau) \cdot \nabla f(r,\tau))$$

which Iijima calls the *basic equation of figure*. This is simply the formulation of the affine linear Gaussian scale-space as a partial differential equation.

In 1981 another Japanese scientist, Nobuki Otsu, wrote his thesis entitled 'Mathematical Studies on Feature Extraction in Pattern Recognition' [161]. He modified Iijima's five one dimensional axioms to derive a two dimensional Gaussian scale-space. He derives a transformation $\tilde{f}$ of an image f such that the axioms in Table 3.2 hold.

Axiom 2 in Table 3.2 implies that the integral kernel is symmetric ($W(r, r' + a) = W(r - a, r')$) and thus it is a convolution kernel, namely,

$$W(r, r') = W(r - r'). \tag{4.4}$$

	Axiom	Formulation
1.	**Linear Integral Operator**	$\exists W : \mathbb{R}^2 \times \mathbb{R}^2 \mapsto \mathbb{R}^2$ such that $\tilde{f}(r) = \int_{\mathbb{R}^2} W(r, r')f(r')dr' \; \forall \; r \in \mathbb{R}^2$
2.	**Translation Invariance**	$\forall \; r \in \mathbb{R}^2, a \in \mathbb{R}^2$, $\tilde{f}(r-a) = \int_{\mathbb{R}^2} W(r, r')f(r'-a)dr'$
3.	**Rotation Invariance of the Kernel**	For any rotation matrix T_Θ, and $\forall \; r = (x, y)^T \in \mathbb{R}^2$, $W(T_\Theta r) = W(r) = W(x^2 + y^2)$
4.	**Separability**	$\exists u : \mathbb{R} \mapsto \mathbb{R}$ such that $W(r) = u(x)u(y)$
5.	**Normalization of Energy**	preservation of nonnegativity: $\tilde{f}(r) \geq 0 \; \forall \; f(r) \geq 0$ average grey level invariance: $\int_{\mathbb{R}^2} \tilde{f}(r)dr = \int_{\mathbb{R}^2} f(r)dr$

Table 4.2: Otsu's Two Dimensional Axioms [161]

From Axioms 3 and 4 in Table 3.2 $W(r) = k \exp\{c(x^2 + y^2)\}$ for some parameters $k, c \in \mathbb{R}$ can be easily derived. Axiom 5 in Table 3.2 implies that $W(r) \geq 0$ and $\int_{\mathbb{R}^2} W(r)dr = 1$ respectively. Using these results and the five axioms he shows that $k = \frac{1}{2\pi\sigma^2}$ and $c = -\frac{1}{2\sigma^2}$ and the Gaussian kernel is obtained,

$$W(r) = \frac{1}{2\pi\sigma^2} \exp\left\{ -\frac{x^2 + y^2}{2\sigma^2} \right\}.$$

Otsu also derives an N-dimensional Gaussian scale-space in his thesis [161]. Taking $\rho = \sigma^2/2$, he starts by defining

$$T(\rho)f(r) = \frac{1}{(4\pi\rho)^{N/2}} \exp\left\{ -\frac{|r|^2}{4\rho} \right\},$$

and then via Fourier techniques obtains

$$\frac{\partial \tilde{f}(r, \rho)}{\partial \rho} = \Delta\big(\exp(\rho\Delta)f(r) \big) = \Delta\tilde{f}(r, \rho)$$

so that $\tilde{f}$ satisfies the isotropic linear diffusion equation.

All these Japanese works only became known from 1997. Before this Witkin's 1983 work was believed to be the starting point of the Gaussian scale-space. We go into this is more detail in this next section.

4.4 The Gaussian Scale-Space

The Gaussian convolution was first represented as a scale-space by Witkin in 1983 [246]. In 1980, though, Stansfield also discusses a Gaussian scale-space idea but without the necessary axioms and mathematical backbone [220]. He applies the idea to designing a commodity expert. In addition, in 1980 Marr published his well-known work 'Theory of Edge Detection' [139], in which he describes using a Gaussian convolution as a smoother and tracks zero-crossings to aid edge detection. Zero-crossings are those points where significant intensity changes are detected. Additional work was done by Crowley in his PhD thesis [44]. He developed the DOLP transform, a class of reversible transforms, and uses the cascading property of the Gaussian (discussed later on) to speed up his algorithm from $O(N^2)$ to $O(NlogN)$. He makes use of a discretized Gaussian though by sampling the domain and constructs a tree-like representation for an image using his transform. Additional attempts have also been published [109, 250, 32, 77, 191, 136, 140].

Wikin's 1983 formulation is as follows. His first formulation is for one-dimensional signals in order to initially develop his ideas. By assuming linearity the integral operator to be used must then involve a family of kernels $\{k_t : t \geq 0\}$ such that $T_t(f)(x) = \int_{\mathbb{R}} k_t(x, x')f(x')dx'$ [239]. By additionally assuming translation invariance, so that $\tau_a T_t = T_t \tau_a \ \forall (a \in \mathbb{R}, t > 0)$ and for a shift operator τ_a, the kernel must be a convolution kernel (Equation 4.4) [239]. The Gaussian convolution is thus argued for based on its 'well-behavedness', namely that it is symmetrical about its mean and decreases away from the mean providing less weight to pixel values further away from the focus pixel. The additional assumption is that zero-crossings of the Gaussian and its derivatives may appear but not disappear as scale decreases. This assumption ensures that the Gaussian is the only convolution kernel which provides the 'well-behavedness' required. Gaussian smoothing is obtained for a continuous signal $f : \mathbb{R} \mapsto \mathbb{R}$ as follows:

- The Gaussian smoothed version of f at scale $t \in \mathbb{R}_+ \backslash \{0\}$ is obtained as the convolution

$$L_f(t)(x) = \int_{-\infty}^{\infty} g_t(\xi)f(x - \xi)d\xi = (g_t * f)(x) \ \forall \ x \in \mathbb{R}, t > 0$$

  where $g : \mathbb{R} \times \mathbb{R}_+ \backslash \{0\} \mapsto \mathbb{R}$ is the one-dimensional Gaussian kernel

$$g_t(x) = \frac{1}{\sqrt{2\pi t}} e^{-x^2/2t}.$$

- The original signal f is defined as the representation at scale 0:

$$L_f(0)(x) = f(x) \; \forall \; x \in \mathbb{R}. \tag{4.5}$$

For every $t > 0$, $L_f(t)$, is called the *scale-space image* of f at scale t. Successive smoothing gradually suppresses fine detail making the signal smoother and more blurred each time.

For an N-dimensional signal $f : \mathbb{R}^N \mapsto \mathbb{R}$ the scale-space representation is similarly obtained using the N-dimensional Gaussian kernel.

- The Gaussian smoothed version of f at scale $t \in \mathbb{R}_+ \backslash \{0\}$ is obtained as the convolution

$$L_f(t)(x) = \int_{\xi \in \mathbb{R}^N} g_t(\xi) f(x - \xi) d\xi = (g_t * f)(x)$$

$\forall \; x = (x_1, x_2, ..., x_N) \in \mathbb{R}^N$ where $g : \mathbb{R}^N \times \mathbb{R}_+ \backslash \{0\} \mapsto \mathbb{R}$ is the N-dimensional Gaussian kernel

$$g_t(x) = \frac{1}{(2\pi t)^{N/2}} e^{-\frac{1}{2t} x^T x}.$$

- The original signal f is defined as the representation at scale 0:

$$L_f(0)(x) = f(x) \; \forall \; x \in \mathbb{R}^N. \tag{4.6}$$

We see that Gaussian smoothing a simply a diffusion process by which the high frequencies are removed. This can be seen easily by applying the convolution theorem [166] as the Fourier transform of the Gaussian remains the Gaussian. The two-dimensional Gaussian scale-space can be derived as the solution of the diffusion equation

$$\frac{\partial L_f(t)(x)}{\partial t} = \frac{1}{2} \frac{\partial^2 L_f(t)(x)}{\partial x^2}$$

with initial condition $L_f(0)(\cdot) = f(\cdot)$, see [109]. As is well-known, this parabolic partial differential equation models the evolution over scale [67, 242, 221]. In its original form (see [30]) it models the flow of heat along a rod length ℓ, say, at time t with initial state $f(x)$ such that along each cross-section the temperature is uniform. The constant on the right hand side is

determined by the heat-conductive properties of the rod material. In higher dimensions the partial differential equation has the form

$$\frac{\partial L_f}{\partial t} = \kappa \bigtriangledown^2 L_f,$$

where κ is the thermal diffusivity as in one dimension and $\bigtriangledown^2$ is the Laplace operator. Koenderink's scale-space derivation, as described in [123], is done for two dimensions but can be reduced to one dimension as well.

The term *scale-space* is reserved for multi-scale representations for which similar theoretical properties can be proven, the most important being that of non-creation of 'new' or 'artificial' structures [123]. We provide a formal definition in Section 4.6. Note that there is a subtle difference between the terms multi-scale and multiresolution, however the terms are used freely and no exact difference is clear.

4.4.1 Gaussian Scale-Space Properties

How the Gaussian kernel smooths a signal

The scale parameter t is the standard deviation in the Gaussian kernel. Thus it acts by averaging the signal symmetrically in every direction with increasing window size as t increases. Structures with support smaller than t will then be suppressed [123].

The smoothness obtained is measured in different ways by different authors. For example, in [250] regularity appears as the convergence of the convolution kernels to the Dirac delta distribution and in [64] as the Fourier transform becoming 1 everywhere. In [12, 61] infinitely differentiable convolution kernels are assumed which are rapidly decreasing functions of x. In [120] the kernels are assumed to be Borel measurable and in [125] the kernels are assumed to converge for $t \to 0^+$ in the L^1 norm to the Dirac distribution. In [4] it is required that for smooth f and g

$$\|L_{f+hg}(t) - (L_f(t) + hg)\|_\infty \leq Cht, \ \forall h, t \in [0, 1]$$

where C may depend on f and g and $L_f(t)$ is the Gaussian convolution of f. In [168] the kernels are assumed to be separately continuous in x and t. More in line with signal and image processing, the LULU smoothers for sequences [183] create smoother versions of their input which are n-monotone if every window of length n is monotone non-increasing or non-decreasing. A similar

definition applies for the LULU smoothers for images, see Section 5.3. Other filters in signal and image processing have similar results.

Of all the possible probability density functions the Gaussian is the one with maximum entropy [12]. Entropy (known as Shannon entropy) is a measure of uncertainty associated with a random variable. With the Gaussian having maximum entropy we are thus making use of a kernel which applies the least amount of prior assumptions and structure onto the signal, as is desired, thereby further enabling smoothing with the Gaussian kernel.

Semigroup and Cascading Property

Since the Gaussian kernel exhibits the semigroup property $g(\cdot,t) * g(\cdot,s) = g(\cdot,t+s)$, a representation at a coarser scale t_2 can be computed from a representation at a finer scale t_1 by an additional convolution with parameter $t_2 - t_1 > 0$ i.e. $L_f(t_2)(\cdot) = (g_{t_2-t_1} * L_f(t_1))(\cdot)$, so that a cascade smoothing property is implied [123].

Separability

The N-dimensional Gaussian kernel $g : \mathbb{R}^N \mapsto \mathbb{R}$ can be written as the product of N one-dimensional Gaussian kernels $g_1 : \mathbb{R} \mapsto \mathbb{R}$,

$$g(x,t) = \prod_{i=1}^{N} g_1(x_i,t), x = (x_1, x_2, ..., x_N) \in \mathbb{R}^N$$

since

$$\frac{1}{(2\pi t)^{N/2}} e^{-\frac{1}{2t}x'x} = \prod_{i=1}^{N} \frac{1}{(2\pi t)^{1/2}} e^{-\frac{x_i^2}{2t}}.$$

This is a useful property, especially for decreasing computational complexity [123].

The Maximum Principle

This property is exactly the strong maximum principle of parabolic equations [158] which states that if a function attains its maximum on the interior of its domain the function is constant. In terms of the scale-space then if $x_0 \in \mathbb{R}$ is a local maximum of $x \mapsto L_f(t_0)(x)$ at a certain scale $t_0 \in \mathbb{R}_+$, then the Laplacian is negative $\bigtriangledown^2 L_f(t_0)(x_0) < 0$ i.e. $\partial_t L_f(t_0)(x_0) < 0$, and if this x_0 is a local minimum then $\bigtriangledown^2 L_f(t_0)(x_0) > 0$ i.e. $\partial_t L_f(t_0)(x_0) > 0$. This means that small local variations are suppressed so that a 'hot spot' will not become warmer and a 'cold spot' not cooler [12, 85, 125].

Scaling Property

If $f(x) = f'(sx)$, let $x' = sx$ and $t' = s^2t$. Then $L'(\cdot, t') = g(\cdot, t') * f'$ and it can be shown that the two representations are the same $L(x,t) = L'(x',t')$ i.e. stretching the parent kernel such that the areas remain the same [168] (see [123] for a proof). Also see [125] where it is shown that this scale invariance follows from the semi-group property when combined with isometry invariance (symmetry) and causality.

Scale-Space Derivatives and Infinite Differentiability

We recall the notation for multi-scale derivatives. Let $n = (n_1, n_2, ..., n_N) \in \mathbb{Z}_+^N, n_i \in \mathbb{Z}_+$, $x = (x_1, x_2, ..., x_N) \in \mathbb{R}^N$ and $x^n = x_1^{n_1} x_2^{n_2} ... x_N^{n_N}$. The

$$\partial_{x^n} = \partial_{x_1^{n_1}} \partial_{x_2^{n_2}} ... \partial_{x_N^{n_N}}$$

is the derivative of order $|n| = n_1 + n_2 + ... + n_N$. The multi-scale derivatives are the scale-space derivatives of f at scale t and are given by

$$\begin{aligned} L_f^{x^n}(t)(x) &= \partial_{x^n} L_f(t)(x) = \left(g_t^{x^n} * f\right)(x) \\ &\quad \text{where } g_t^{x^n} \text{ is the partial derivative of the Gaussian} \\ &\quad \text{kernel of order } |n| \\ &= \int_{x' \in \mathbb{R}^N} g_t^{x^n}(x - x') f(x') dx' \\ &= \int_{x' \in \mathbb{R}^N} g_t^{x^n}(x') f(x - x') dx'. \end{aligned}$$

The scale-space derivatives are guaranteed to converge for any $t > 0$ if f is bounded above by some polynomial. Since the Gaussian function decreases exponentially if there exists $c_1, c_2 \in \mathbb{R}_+$ such that $|f(x)| \leq c_1(1 + x'x)^{c_2}$ then even if f is not differentiable convergence is guaranteed. The convolution provides a strong regularizing property and for every $t > 0$ the scale-space derivatives can be treated as infinitely differentiable [123].

In addition the scale-space properties mentioned thus far transfer to the scale-space derivatives as well. Namely, they satisfy the diffusion equation, also get successively smoother, ensure non-enhancement of extrema and possess the cascading smoothing property. They satisfy a scaling property but one which is slightly different,

$$g_{x^n}(x,t) = s^{N+|n|} g_{x^n}(sx, s^2t).$$

We then have that $L_f^{x^n}(t)(x) = \partial_{x^n} L'_f(t')(x') = s^{|n|} L_f'^{x'^n}(t')(x')$. The coordinates can however by normalized to $\xi = x/\sqrt{t}$ and $\xi' = x'/\sqrt{t'}$ to make them dimensionless and then $L_f^{\xi^n}(t)(x) = L_f'^{\xi'^n}(t')(x')$.

Other Properties [123]

For a function h with Fourier transform $\hat{h}$ the normalized second moments Δx and $\Delta\omega$ in the spatial and Fourier domain, which describe the spread of the distribution of these two functions, are

$$\Delta x = \frac{\int_{x\in\mathbb{R}} x'x|h(x)|^2 dx}{\int_{x\in\mathbb{R}} |h(x)|^2} dx \text{ and } \Delta\omega = \frac{\int_{\omega\in\mathbb{R}} \omega'\omega|\hat{h}(\omega)|^2 d\omega}{\int_{\omega\in\mathbb{R}} |\hat{h}(\omega)|^2} d\omega.$$

The uncertainty relation states that $\Delta x \Delta\omega \geq \frac{1}{2}$ and the Gaussian kernel is the only real kernel that gives equality here. The Gaussian kernel is also the only rotationally symmetrical kernel that is separable in Cartesian coordinates.

4.4.2 Gaussian Scale-Space Axioms for Uniqueness

A number of authors have, since Witkin and Koenderink's work, made similar derivations of the Gaussian scale-space and its uniqueness based on various sets of axioms. The main idea throughout all the research done is that the smoothing mechanism does not allow creation of spurious structures. This idea has been formulated in various works. We discuss them now. A summary table is presented in Table 4.4.2 (replicated from [239]). Note, that the uniqueness referred refers to the Gaussian kernel in the convolution formula, not the unique existence of only the Gaussian scale-space.

Witkin 1983 [246]

Witkin introduced the theory for one-dimensional signals and observed that new local extrema were not created. This property extends to the scale-space derivatives and he thus tracked the zero-crossings across scale forming a tree data structure for the signal. He mentions a link between the length of the branches of the tree and the perceptual saliency of the viewer.

Axiom	I1	I2	I3	O	K	Y	B	L1	F1	A	P	N	L2	F2
Convolution Kernel	•	•		•		•	•	•	•	•	•		•	•
Semigroup Property	•	•						•	•	•	•	•	•	•
Locality										•				
Regularity						•	•	•	•	•	•		•	•
Infinitesimal Generator											•			
Maximum Loss Principle			•											
Causality					•	•	•	•					•	
Nonnegativity	•	•		•						•	•			•
Tikhonov Regularization												•		
Average Grey Level Invariance			•	•			•	•		•	•			
Flat Kernel for $t \to \infty$						•			•					
Isometry Invariance (symmetry)		•		•		•		•	•	•	•	•	•	•
Homogeneity and Isotropy					•									
Separability				•					•					
Scale Invariance	•	•				•	•		•		•	•		•
Valid for which dimensions?	1	2	2	2	1,2	1,2	1	1	> 1	N	1,2	N	N	N

Table 4.3: Comparison of the Gaussian Scale-Space Axioms [239]. (Key: I1 = [97][98][87][88], I2 = [98][87]), I3 = [89], O = [161], K = [109], Y = [250], B = [12], L1 = [120], F1 = [64], A = [4], P = [168], N = [157], L2 = [125], F2 = [61]

Koenderink 1984 [109]

Koenderink shows that the family generated by using the Gaussian convolution is unique when assuming axioms of causality, homogeneity and isotropy. These are, more specifically, that 'spurious events' may not be generated so that every feature at a coarse scale level must have a 'cause' at a finer scale level (every isophote - constant luminosity levels - in scale-space must be upwards convex), and that smoothing is both scale and spatially invariant. Using these he shows that the scale-space representation must satisfy the diffusion equation and since the Gaussian kernel is the Green's function of the diffusion equation the uniqueness of the solution follows. Green's function is a function used to show existence and uniqueness of the solution of inhomogeneous differential equations [10]. Since the scale-space derivatives also satisfy the diffusion equation the property of no new zero-crossings with increasing scale still holds.

Yuille and Poggio 1983 [250]

Yuille and Poggio impose their assumptions on the filter F used as boundary conditions in two dimensions. Their assumptions are as follows.

1. **The filter is shift-invariant:** The filter is therefore a convolution $F * f = \int F(\underline{x} - \underline{\xi}) f(\underline{\xi}) d\underline{\xi}$.
2. **The filter has no preferred length**
3. **The filter covers the entire image at sufficiently small scales:** $\lim_{t \to 0} F(\underline{x}, t) = \delta(\underline{x})$ where $\delta(\underline{x})$ is the Dirac delta function.
4. **The position of the center of the filter is independent of** t
5. **A Flat kernel as** $t \to \infty$**:** As $|\underline{x}| \to \infty$ and $t \to \infty$ we have that the filter goes to 0 and so $\lim_{t \to \infty} k_t(x) = 0$.

Note that symmetry is not one of their requirements. With these assumptions they are able to prove that in one and two dimensions the Gaussian filter is the only filter which doesn't create zero-crossings as scale increases, and in two dimensions, when using the directional operator along the gradient, there is no filter which obeys their assumptions and does not create zero-crossings as scale increases.

A related uniqueness formulation is also presented in [85].

Witkin et al 1986 [12]

Witkin et al prove the uniqueness of the Gaussian kernel in one dimension under a number of conditions, the main one being a monotonicity condition such that zero-crossings appear from coarse to fine scale but existing ones never disappear. This means that the local maxima (and minima) of the surface swept out by f always increase (and decrease) as scale increases so that peaks and valleys become more pronounced as scale increases. Their additional assumptions in order to prove the uniqueness are that the kernel g is infinitely differentiable and rapidly decreasing (Schwartz), there exists a kernel h such that $g(x,t) = th(xt)$ so that the scale parameter t stretches the kernel along the x-axis while keeping its area invariant, the kernel is symmetric, that is $g(x,t) = g(-x,t)$, the kernel is normalized so that $\int_{-\infty}^{\infty} g(u,t)du = \int_{-\infty}^{\infty} h(v)dv = 1$, and there exists a $p \in \mathbb{Z}$ such that $h^{(2p)}(0) \neq 0$, that is, not all derivatives of h vanish at 0. The normalization assumption insures that if f is a constant signal then it remains the same constant though the convolution. The authors also show that the diffusion equation is equivalent to requiring the monotonicity condition.

In two dimensions the zero-crossings are more complicated. They do not vanish as scale increases but can split and merge.

Florack et al 1992 [64]

Florack et al also prove that the Gaussian kernel is unique. They use the assumptions of linearity, spatial shift invariance, isotropy and scale invariance as the basic axioms, and then derive a weak semi-group property which ensures that several successive scalings is the same as performing a single equivalent scaling and combine it with a uniform scaling property over scales to finally show the uniqueness.

Lindeberg 1994 [123]

In Lindeberg's 1994 book all his work over the previous decade is nicely summarized. He uses non-creation of features as well as a semi-group structure to

prove the uniqueness of the Gaussian kernel (proven in his 1990 paper [120]). He also shows that the number of zero-crossings in the second derivative decreases monotonically with scale.

In a later paper by Lindeberg [124] the main results of his book are summarized.

Alvarez et al 1993 [4]

Alvarez et al present a very theoretical paper on the requirements of an image processing transform. They classify the requirements as either *architectural*, *stability* or *morphological*. The architectural axioms are those of recursivity (semi-group property, causality), existence of an infinitesimal generator (to remove the dependence on h, the sampling distance), regularity

$$\|L_{f+hg}(t) - (L_f(t) + hg)\|_\infty \leq Cht \ \forall h, t \in [0,1],$$

for smooth f, g where C depends on f, g, and locality, namely, for small t, $L_f(t)$ at any point x is determined by its vicinity, namely, for all $f, g \in C^\infty$ whose derivatives are equal at x,

$$(L_f(t) - L_g(t))(x) = o(t) \text{ as } t \to 0^+.$$

The stability axioms boil down to the comparison principle i.e. no enhancement can be made. This is also interpreted by [239] as nonnegativity, that is, $k_t(x) \geq 0 \ \forall x, \ \forall t > 0$, to ensure new level crossings do not appear. This is satisfied if we require monotonicity,

$$f \leq g \longrightarrow L_f(t) \leq L_g(t) \ \forall t > 0,$$

or preservation of non-negativity,

$$f \geq 0 \longrightarrow L_f(t) \geq 0 \ \forall t > 0.$$

The morphological axioms are average grey level invariance, translation invariance, isometry invariance and scale invariance:

AGLI: $\int_{\mathbb{R}^N} L_f(t)(x)dx = \int_{\mathbb{R}^N} f(x)dx \ \forall t > 0$. This requires that the kernels be normalized $\int_{\mathbb{R}^N} k_t(x)dx = 1$ or that grey level shift invariance is satisfied

$$L_f(t)(0) = 0, L_{f+c}(t) = L_f(t) + c.$$

TI: $L_{\tau_h f}(t) = \tau_h(L_f(t))$ where $\tau_h f = f(x+h)$.

II: $L_{Rf}(t) = RL_f(t)$ for all orthogonal transforms R defined by $(Rf)(x) = f(Rx)$.

SI: For any λ and t, there exists t' such that $D_\lambda L_f(t') = L_f(t)D_\lambda$ so that the result of $L_f(t)$ is independent of the size of the features involved.

They show that a sequence of multi-scale operators $L_f(t)(x) = u(t,x)$ is a solution of a second order partial differential equation

$$\frac{\partial u}{\partial t} F(D^2 u, Du) \text{ with } u(0,x) = f(x)$$

with certain requirements satisfied, namely recursivity, regularity, locality, translation and shift invariance. The heat equation is then the only linear isometrically invariant special case of this

They in addition combine the multi-scale ideas of Witkin et al (Gaussian scale-space and the heat equation) with the morphology scale-space ideas (structuring elements of differing sizes and the opening and closing operations) to obtain a 'class of morphological multi-scale analyses'. These satisfy

$$\frac{\partial u}{\partial t} \beta(t\text{curv}(u))|Du|$$

where β is an arbitrary non-decreasing real function and curv(u) is the curvature of the level set of u passing through x. This combination keeps the noise-elimination properties of the heat equation but is now shape-preserving due to the morphological operators.

Pauwels et al 1995 [168]

In this well-written 1995 paper by Pauwels, it is described how by assuming a semi-group property (what they and [4] call recursivity) and scale-invariance, and other more trivial assumptions, it is possible to derive a class of scale-space operators which depend on a parameter α for which the Gaussian is a special case when $\alpha = 2$.

They begin with assuming that the operators are linear, as all the other authors do as well, and are integral operators. This also allows operations to be run in parallel as as comparisons of neighbouring pixels are done.

So the operator has the form $(L_f(t)(x) = \int_{\mathbb{R}} k_t(x,\xi)f(\xi)d\xi$ where k_t is the integral kernel. By also assuming shift-invariance the kernel must then be a convolution kernel i.e. $k_t(x,t) = k_t(x-t)$, and so $K_t = k_t * f$. They impose the following conditions on the kernel k_t:

- k_t **is mass-preserving:** $\int k_t(x)dx = 1$ so that $k_t * 1 = 1$ and a constant signal is not changed.
- k_t **is even:** $k_t(x) = k_t(-x)$
- k_t **is integrable** ($k_t \in L^1$)**:** otherwise the convolution is not well-defined
- k_t **is a continuous function of** t **and** x

Then assuming an additive property, namely recursivity: $K_0(f) = f$ and $K_t K_s = K_{t+s}\ \forall t, s \geq 0$ (the kernel also forms such a semi-group: $k_t * k_s = k_{t+s}$) and scale-invariance they derive a rescaling of this kernel family from a fixed kernel ϕ which depends on a parameter α. Thus recursivity and scale-invariance are not sufficient to single out the Gaussian kernel as unique as it is a special case when $\alpha = 2$. They obtain this same result for two dimensions. They delve deeper and show that the Gaussian kernel is only unique if requiring the existence of an infinitesimal generator of differential form. Then the α's can only be even integers and only for $\alpha = 2$ do we obtain positivity everywhere.

Nielson et al 1996 [157]

In this paper scale-space, functional minimization and edge detection filters are compared. They show that the Gaussian scale-space can be obtained through Tikhonov regularization if requiring scale invariance and a semi-group constraint (recursivity). Regularization is the minimization of a signal with respect to an energy functional. A function u is a Tikhonov regularization of a signal $f \in L^2(\mathbb{R}^2)$ if it minimizes the energy function

$$E_f[u] = \int_{\mathbb{R}} \left[(f-u)^2 + \sum_{i=1}^{\infty} \lambda_i \left(\frac{d^i u}{dx^i} \right)^2 \right] dx$$

where $\lambda_i \geq 0$. They also show that this regularization then further more results in the heat equation. Their results are also proven for higher dimensions.

Florack 1996 [61]

In this paper Florack presents a formal theoretical definition of an image with an associated filter space as well as group structure. He also shows in this manner, that the Gaussian is the unique filter for a linear convolution integral operator.

Relation to the Japanese Gaussian Scale-Space Axioms [239]

The Japanese axioms for uniqueness differ from the more recent approaches in two ways. Firstly, the earlier Japanese approaches use less axioms than even recent approaches. Secondly, the axioms are simpler as they don't require any Fourier analysis, complex integrals nor functional analysis.

4.4.3 Discretizing the Gaussian Scale-Space

In practice signals are not continuous. We only have discrete data when a signal, image or video is captured. A signal is captured as a discrete sequence, an image as a matrix, and a video as a discrete sequence of matrices. The Gaussian scale-space theory presented up to now has assumed a continuous input f. The actual implementation of the continuous Gaussian scale-space thus proves difficult. There are two options presented in [120, 121, 123].

The first option is the obvious one, namely, the sampled application of the continuous theory. More specifically this involves discretizing the developed continuous theory and the equations therein via numerical methods. This can be done relatively effectively by using sampled values of the Gaussian kernel together with the rectangle rule of integration. This method, although it gives accurate numerical results, does not guarantee the non-creation of structure as scale increases, which is the most important requirement for a scale-space. The discretization of the diffusion equation is also an option. This is proposed and done with the ordinary 5-point Laplace operator thereby forming a set of ordinary differential equations. We will return to the discretized diffusion equation after we first deal with the second option. The scale, currently continuous, should also be discretized in a logical manner to enable the application. This will be discussed later in this chapter.

In order to maintain the desired theoretical structure of the continuous theory

through the discretization process, the second option is to develop an entirely new (discrete) theory based on the same axioms but modified for the discrete structure we must now work with. This method in fact gives a computational advantage over the first option as well. A discrete convolution of f by a kernel T, namely $T(\cdot,t) * f(\cdot)$, is obtained as

$$L_f(t)(x) = \sum_{n=-\infty}^{\infty} T(n,t) f(x-n), t > 0.$$

The scale parameter t will be kept continuous though to allow for the freedom of choosing any scale t greater than zero instead of only certain values.

Lindeberg first develops this new theory for one dimension. His main requirement for the discrete kernel, $T(n,t)$, is that the number of local extrema in the convolved signal does not exceed the number of local extrema in the original signal. This implies that the amount of structure in the signal will decrease as scale increases, as is the case with the continuous theory. He calls a kernel which satisfies this property as a *scale-space kernel*. He then derives the discrete scale-space as

$$\begin{aligned} L_f(0)(x) &= f(x) \\ L_f(t)(x) &= \sum_{n=-\infty}^{\infty} T(n,t) f(x-n), t > 0 \end{aligned}$$

where $T(n,t) = e^{-t} I_n(t)$ and I_n is the modified Bessel functions of integer order which was discussed in the earlier part of this chapter. This discrete scale-space satisfies the following properties:

- The amount of structure does not increase with scale so that for $t_2 > t_1$ the number of local extrema in $L_f(t_2)(x)$ is not more than the number in $L_f(t_1)(x)$.
- A semi-group property: $L_f(t_2)(\cdot) = T(\cdot, t_2 - t_1) * L_f(t_1)(\cdot)$.
- Normalization: $\sum_{n=-\infty}^{\infty} T(n,t) = 1$.
- Symmetry: $T(-n,t) = T(n,t)$.
- $var\,(T(\cdot,t)) = \sum_{n=-\infty}^{\infty} n^2 T(n,t) = t$.

There are a few points to consider for numerical implementation of this discrete scale-space. Firstly, the infinite sum needs to be replaced by a finite

one, that is, we sum from $n = -N$ to N for some finite N. This N can be chosen such that the absolute error in L due to the truncation does not exceed a given error limit ϵ_{trunc}. Secondly, the modified Bessel function need to be calculated with the recurrence relation,

$$I_{n-1}(t) - I_{n+1}(t) = \frac{2n}{t} I_n(t)$$

which is stable for backward iteration. In Lindeberg's work he states that built in routines are not available to evaluate the I_n's and hence his development of code making use of this recurrence relation. However, at present this is no longer true. In Mathematica the function *BesselI[n,t]* is available, in R the function *besselI[n,t,expon.scaled=FALSE]* is available, in SAS the function *IBESSEL(t,n,0)* is available, and in MATLAB the function *besseli(t,n)* is available, to name a few. This is thus no longer a major problem for the implementation.

We return to investigating the discretization of the diffusion equation. The convolution $L_f(t)(x)$ above is the solution of the following partial differential equation

$$\frac{\partial L_f(t)(x)}{\partial t} = \frac{1}{2}\left(L_f(t)(x+1) - 2L_f(t)(x) + L_f(t)(x-1) \text{ for } x \in \mathbb{Z}\right).$$

A two-dimensional discrete scale-space is more tricky to develop since the non-creation of structure as scale increases isn't always true in two dimensions. Lindeberg requires instead that local extrema must simply not be enhanced as scale increases, that is local maxima must not increase and local minima not decrease as scale increases. This reduces to the one dimensional axiom if the space is reduced to one dimension. He derives the scale-space operator as

$$L_f(t)(x,y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} T(m,n,t) f(x-m, y-n), t > 0$$

which satisfies the differential equation

$$\frac{\partial L_f(t)(x,y)}{\partial t} = \frac{1}{2}\left((1-\gamma) \bigtriangledown_5^2 L_f(t)(x,y) + \gamma \bigtriangledown_\times^2 L_f(t)(x,y)\right)$$

where $\bigtriangledown_5^2$ is the five-point operator

$$(\bigtriangledown_5^2 f)(x,y) = f(x-1,y) + f(x+1,y) + f(x,y-1) + f(x,y+1) - 4f(x,y)$$

and $\bigtriangledown^2_\times$ is the cross operator

$$\begin{aligned}(\bigtriangledown^2_\times f)(x,y) &= \frac{1}{2}\left(f(x-1,y-1)+f(x-1,y+1)+f(x+1,y-1)\right.\\ &\left.+f(x+1,y+1)-4f(x,y)\right),\end{aligned}$$

both approximations of the two-dimensional Laplace operator $\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}$. The kernel T is symmetric, that is, $T(-x,y,t)=T(x,y,t)$ and $T(y,x,t)=T(x,y,t)$ and satisfies a continuity property

$$\|T(\cdot,\cdot,t)-\delta(\cdot,\cdot)\|_1 \to 0 \text{ as } t \to 0$$

where δ is the two-dimensional delta function which is 1 at $(0,0)$ and 0 elsewhere. The operator L_f is linear, shift-invariant and satisfies the semi-group property. If $\gamma = 0$ then $T(m,n,t)$ is a separable convolution kernel and so

$$L_f(t)(x,y) = \sum_{m=-\infty}^{\infty} T(m,t) \sum_{n=-\infty}^{\infty} f(x-m,y-n), t>0$$

where $T(n,t)=e^{-t}I_n(t)$. In addition if (x_0,y_0) is a local maximum (minimum) point then

$$\frac{\partial L_f(t)(x_0,y_0)}{\partial t} \leq (\geq)\ 0.$$

Lindeberg defines (x,y) as a local maximum (minimum) point if for $f:\mathbb{Z}^2 \mapsto \mathbb{R}$, we have $f(x,y) \geq (\leq) f(\xi,\eta)\ \forall\ (\xi,\eta) \in N_8(x,y)$ where $N_8(x,y)$ defines the eight vertical, horizontal and diagonal neighbours of the point (x,y). This two dimensional formulation can also be generalized to higher dimensions, see [123, Chapter 4].

4.4.4 Relating Scales

Having thus obtained a scale-space of the signal f the true question now is how do we use all these smoothed versions of the signal as one? How do we construct links between the scale levels? Witkin [245] presents a tree structure in this regard however as stated in [4] this method implies heavy implementation from the computational point of view and is unstable because of the follow-ups to check for edges (zero-crossings) at each scale. However, with today's computing power this statement may no longer be valid. Relating the scales is directly related to feature detection via the scale-space, so we'll return to this in Chapter 4.8.2.

4.5 Other Scale-Spaces Developed

The Gaussian scale-space provides a multi-scale representation of an image such that a full image is derived at each possible (or required) scale level [123]. This is in contrast to a pyramid representation in which the original image size is reduced at every step and provides then a multiresolution representation. Such a multiresolution technique provides reduced computational requirements but does not allow for explicit access to each of the scale levels. Even more importantly, it does not allow for a method of associating structures over the scale levels, and for which a scale-space does. Thus we focus on multi-scale methods.

Numerous researchers have introduced multi-scales methods different to the Gaussian scale-space. We discuss them now. Note the the uniqueness of the Gaussian scale-space is specific to the axioms imposed.

Scale-Space via the Gabor Functions

Daniel Gabor suggested the Gabor functions in 1946 [68] when Fourier analysis didn't provide him with the freedom to vary the frequency parameter through time. The Gabor functions are as follows,

$$g_{\ell,n}(x) = g(x - a\ell)e^{2\pi ibnx}, -\infty < \ell, n < \infty$$

where $g \in \mathcal{L}^2(\mathbb{R})$ and $\|g\| = 1$, i.e. they are a family of functions built from translations and modulations of a function g. By choosing the function g to be the Gaussian, a scale parameter is introduced and a hierarchical decomposition of a signal can be obtained.

In [134] the Gabor functions are used to develop a time-frequency dictionary of functions $g_\gamma(t)$ to yield an adaptive decomposition of a signal f, namely,

$$f(t) = \sum_{n=-\infty}^{\infty} a_n g_{\gamma_n}(t),$$

in which the functions are selected in order to best match the structure of the signal. The possibility of applying this to signal coding is discussed since it will provide a more efficient coding than orthogonal decompositions. In [72] Granlund designs a general parallel and hierarchical operator and bases it in the Gabor functions with g as the Gaussian. His basic idea is for the operator to describe the image locally as a vector with two components, direction and magnitude. In [149] a hierarchical model using multi-oriented, multi-scale

Gabor functions is presented which models the human visual cortex. The model is used for multi-class object recognition by extracting a feature set representing the salient characteristics of the objects.

Nonlinear Anisotropic Diffusion

In 1984 Cohen and Grossberg [37] discuss the diffusion of boundary feature information for a boundary-completion process in the HVS and provide a nonlinear diffusion equation to model the activity, but in 1990 Perona and Malik [170] presented the anisotropic diffusion scale-space in order to improve on the non-meaningful and blurred edges resulting from the Gaussian scale-space. Instead of the constant diffusion coefficient c in Koenderink's linear diffusion equation $I_t = c(I_{xx} + I_{yy})$ [109] they use a coefficient $c(x, y, t)$ dependent on the spatial and scale parameters thereby introducing the nonlinear equation $I_t = c(x, y, t)(I_{xx} + I_{yy}) + \nabla c \cdot \nabla I$. They apply the scale-space for improved edge detection. Whitaker and Pizer [241] combine the information over the scales effectively for edge detection. Shah [216] investigates using nonlinear diffusion for improved segmentation. Alvarez et al [4] discuss Perona and Malik's nonlinear anisotropic equation as well as their own adapted nonlinear approach which is linked with a morphology approach.

Mathematical Morphology

Scale-spaces are also prominent in mathematical morphology. They result from the recursive applications of morphological operators. Some examples follow.

Maragos [135] investigates the morphological scale space using morphological openings and closings which ensure the preservation of edges. Braga-Neto [20] defines a σ-connected operator, that is, an operator connected at scale σ. He uses these operators to obtain a morphological scale-space representation and applies it for automatic target detection. Braga-Neto and Goutsias [26] use greyscale connectivity, namely a grayscale image is connected if all level sets below a pre-specified threshold are connected, to build a morphological scale-space. The apply the scale-space to object extraction, segmentation, and object-based filtering. Brag-Neto [21] also investigates a nonlinear pyramidal image representation scheme via multiscale grain filters by gradually removing connected components from an image that fail to satisfy a given criterion.

4.6 Scale-Space: A Formal Definition

In [20] a scale-space is referred to as a representation which allows for the tracking of the evolution of image structures (e.g. regional maxima and minima) through a continuous range of scales, from fine to coarse, basically an ordered set of derived images which represent the original at alternative scales [123]. These descriptions are very vague although clear as to their intention. We proceed to define a scale-space formally. Let Ω be an infinite space (for example $\mathbb{R}^n$ or $\mathbb{Z}^n$) and $\mathcal{A}(\Omega)$ the set of all real functions defined on Ω. The space Ω is purposefully general so as to provide an axiomatic definition of a scale-space.

First we provide an axiomatic definition for a scaling operator. This definition makes allowance for any domain, discrete, continuous or otherwise.

Definition 28 *An operator $\varphi : \Omega \mapsto \Omega$ is called a* scaling operator *if it is 1) an order preserving mapping, and 2) $\forall\ x \in \Omega$ there exists $a_x \in \Omega$ such that $\varphi^{-1}(x) = \varphi^{-1}(0) + a_x$.*

We now define a measure of smoothness.

Definition 29 *A function $S : \mathcal{A}(\Omega) \mapsto \mathcal{A}(\Omega)$ is called a* measure of smoothness *if the following axioms hold for any $f, g \in \mathcal{A}(\Omega)$:*

A1 $Sf = 0 \iff f$ *is constant.*

A2 $S(\alpha f) = |\alpha| S(f)$

A3 $S(f + g) \leq S(f) + S(g)$

A4 $S(f \circ E_\alpha) = Sf$ *for* $\alpha \in \Omega$ *(translation invariance)*

A5 $S(f \circ \varphi) = Sf$ *(scale invariance)*

In Definition 29 for $\alpha \in \Omega$ the operator $E_\alpha : \mathcal{A}(\Omega) \mapsto \mathcal{A}(\Omega)$ is a shift operator, namely, $(E_\alpha f)(x) = f(x - \alpha)$, and the function $\varphi : \Omega \mapsto \Omega$ is a scaling operator as in Definition 28. Note that the first three axioms for the smoothing operator in Definition 29 are those for a semi-norm. Axioms 4 and 5 are invariance properties. Note also that the operator S is actually a measure of

'roughness' since a larger value indicates less 'smoothness'. As mentioned in [81], the choice of S in general depends on the requirements of the specific task so Definition 29 sets general axioms. A number of alternatives for S have been suggested in literature. For example, in [234] smoothness is considered as a measure of how each data point is similar to or well supported by the data points in its vicinity. Qi and Sun [175] consider a function smooth if it is continuously differentiable, that is the function as well as its first derivative are continuous. We could also consider a function smooth provided the derivatives up to a specific order are continuous, choosing the specific order based on the task at hand.

We now define a scale-space operator.

Definition 30 *Let $\Lambda \subset \mathbb{R}^+$ be the an unbounded set of scale parameters. An operator $\mathcal{L}(f, \lambda) : \mathcal{A}(\Omega) \times \Lambda \to \mathcal{A}(\Omega)$ where $f \in \mathcal{A}(\Omega)$ is a* scale-space operator *if it satisfies the following axioms:*

A1 $\mathcal{L}(f, 0) = f$

A2 *For every $\lambda_1, \lambda_2 \in \Lambda, \lambda_1 < \lambda_2$ we have $S(\mathcal{L}(f, \lambda_2)) \leq S(\mathcal{L}(f, \lambda_1))$. Moreover,*

$$\lim_{\lambda \to \infty} S(\mathcal{L}(f, \lambda)) = 0.$$

A3 $\mathcal{L}(\alpha f, \lambda) = \alpha \mathcal{L}(f, \lambda)\ \forall \alpha > 0$ *(***Positive Homogeneity***)*

A4 *For every $\lambda_1, \lambda_2 \in \Lambda, \lambda_1 < \lambda_2$, there exists an operator $\mathcal{M}(\lambda_1, \lambda_2) : \mathcal{A}(\Omega) \mapsto \mathcal{A}(\Omega)$ such that $\mathcal{M}(\lambda_1, \lambda_2) \circ \mathcal{L}(f, \lambda_1) = \mathcal{L}(f, \lambda_2)$. (***Cascading Property***)*

A5 $E_\alpha \circ \mathcal{L}(f, \lambda) = \mathcal{L}(f, \lambda) \circ E_\alpha$ *(***Translation Invariance***)*

A6 *For each $\lambda \in \Lambda$ there exists $\lambda' \in \Lambda$ such that $\mathcal{L}(f, \lambda') \circ \varphi = \mathcal{L}(f \circ \varphi, \lambda)$ (***Scale Invariance***)*

Some points to take note of. Axiom A1 ensures that the original image forms part of the scale-space. Axiom A4 enables the successive smoothing by $\mathcal{L}(f, \cdot)$ first at a scale λ_1 and then at scale $\lambda_2 > \lambda_1$ on the already smoothed $\mathcal{L}(f, \lambda_1)$. Notice also that $\mathcal{L}$ need not necessarily be linear. In [123, Chapter 3] some general axioms are presented for a linear scale-space ensuring the smoothing operation is a convolution.

For convenience we denote $\mathcal{L}(f, \lambda)$ as $\mathcal{L}_f(\lambda)$ since the first parameter f is fixed and the second parameter λ varied in applications.

Following Definition 30, we define a precise definition of a scale-space associated with a given function $f \in \mathcal{A}(\Omega)$ as the range of the operator $\mathcal{L}_f$.

Definition 31 *Let $f \in \mathcal{A}(\Omega)$. The set*

$$\mathcal{S}_{f,\Lambda} = \{(\lambda, \mathcal{L}_f(\lambda)) : \lambda \in \Lambda\}$$

is called a scale-space *of f generated by the operator $\mathcal{L}$ with respect to scale parameter set Λ and measure of smoothness $S \in \mathcal{A}(\Omega)$.*

In the literature the term scale-space is used with more broad meaning. In addition to the set in Definition 31 the term is also referred to its subsets or to the operator $\mathcal{L}$. As this may lead to confusion, we will use it here only with the meaning given in Definition 31.

We show that the Gaussian scale-space satisfies the axioms of Definition 30 and 31.

Theorem 32 *The Gaussian scale-space operator defined in Section 4.4 satisfies the axioms of Definition 30.*

Proof
For the Gaussian scale-space the scale parameter set Λ is continuous and is given by $\{t : t \geq 0\}$.

A1 This follows from Equations 4.5 and 4.6.

A2 For the Gaussian scale-space operator the measure of smoothness $S \in \mathcal{A}(\Omega)$ defined in Definition 29 is the continuous total variation, namely

$$TV(f) = \int_\Omega |\nabla f(x)| dx.$$

It is clear that $TV(f)$ satisfies the axioms of Definition 29. Since the derivative of the Gaussian scale-space operator is

$$\frac{\partial}{\partial x}\mathcal{L}_f(t)(x) = \left(\frac{\partial}{\partial x} g(x,t)\right) * f(x),$$

and it is well known that the Gaussian density function flattens out as the variance t increases, namely g satisfies for $t_1 < t_2$

$$\frac{\partial}{\partial x}g(x,t_2) \leq \frac{\partial}{\partial x}g(x,t_1),$$

we know that

$$|\nabla \mathcal{L}_f(t_2)(x)| \leq |\nabla \mathcal{L}_f(t_1)(x)|$$

so that $TV(\mathcal{L}_f(t_2)) \leq TV(\mathcal{L}_f(t_1))$. Also as $t \to \infty$ $\frac{\partial}{\partial x}g(x,t) \to 0$ so that $\frac{\partial}{\partial x}\mathcal{L}_f(t)(x) \to 0$ and so

$$\lim_{t\to\infty} TV(\mathcal{L}_f(t)) = 0.$$

A3 This follows immediately for a convolution.

A4 The cascading property of the Gaussian scale-space operator is as follows for $t_2 > t_1$, [123, Chapter 2.4.4]

$$\mathcal{L}(\cdot,t_1) = g(\cdot,t_2-t_1) * \mathcal{L}(\cdot,t_1).$$

So the operator $\mathcal{M}(t_1,t_2)$ is given by a convolution with a Gaussian kernel with parameter $t_2 - t_1$.

A5 Translation-invariance is a required property of the Gaussian scale-space operator [123].

A6 In [123, Chapter 2.4.8] it is verified that for each $t \in \Lambda$ there exists $t' \in \Lambda$ such that $\mathcal{L}(f,t)(x) = \mathcal{L}(f',t')(x')$ where $f \circ \varphi(x) = f'(sx)$, $t' = s^2t$ and $x' = sx$ for $s \in \mathbb{R}^+$.

■

4.7 The LULU Scale-Space

The DPT forms a scale-space in the sense of Definitions 30 and 31 when applied to a function f. We shall prove this. Firstly note that due to the idempotence of the LULU operators $\mathcal{L}_f(\lambda) = \mathcal{L}_{\mathcal{L}_f(\lambda_1)}(\lambda)$, indicating it doesn't really make sense to apply $\mathcal{L}_f(\lambda_1)$ first as the same is achieved by applying $\mathcal{L}_f(\lambda)$ for $\lambda > \lambda_1$. However, the information which is peeled off by first $\mathcal{L}_f(\lambda_1)$ and then $\mathcal{L}_f(\lambda)$ indicates the reason for applying them step by step. Total variation as defined in Definition 2.15 is a smoothing operator as described in Definition 29. The five axioms are proved in [52].

Theorem 33 *The Discrete Pulse Transform, when applied to* $f : \mathbb{Z}^d \to \mathbb{R}$*, derives a scale-space* $S_f^{LULU} = \{(n, P_n(f)) : n \in \Lambda_0 = \{0, 1, 2, ..., N\}\}$ *as described by Definition 31 which we call the* LULU scale-space.

Proof We proceed to show that the axioms in Definition 30 are satisfied by the LULU scale-space.

A1 Since the DPT is the result of P_n, $n = 1, 2, ..., N$, where N is the total number of data points, it is trivial then to have $P_0(f) \equiv f$.

A2 By Theorem 14 we know $P_n(f)$ is total variation preserving so for $n_2 > n_1$

$$\begin{aligned} TV(P_{n_1}) &= \sum_{n=n_1+1}^{N} TV(D_n(f)) \\ &\leq \sum_{n=n_2+1}^{N} TV(D_n(f)) \\ &= TV(P_{n_2}(f)). \end{aligned}$$

Since $D_N(f)$ is constant we know that

$$\lim_{n \to N^+} TV(P_n(f)) = 0.$$

A3 Axiom A3 holds as discussed in detail in Chapter 1 and presented in Theorem 26.

A4 Due to the idempotence of the LULU operators

$$\begin{aligned} \mathcal{L}_{\mathcal{L}_f(n_1)}(n_2) &= P_{n_2}(\mathcal{L}_{\mathcal{L}_f(n_1)}(n_2 - 1)) \\ &= P_{n_2} \circ ... \circ P_{n_1+1}(\mathcal{L}_{\mathcal{L}_f(n_1)}(n_1)) \\ &= P_{n_2} \circ ... \circ P_{n_1+1}(P_{n_1} \circ P_{n_1}(\mathcal{L}_f(n_1 - 1)) \\ &= P_{n_2} \circ ... \circ P_{n_1+1}(P_{n_1}(\mathcal{L}_f(n_1 - 1)) \text{ by idempotence} \\ &= P_{n_2} \circ ... \circ P_{n_1+1}(\mathcal{L}_f(n_1)) \\ &= \mathcal{L}_f(n_2) \end{aligned}$$

A5&A6 These axioms of translation and scale in variance follow immediately from the properties of a separator given in Definition 2 since the LULU operators are separators.

■

Definition 34 *The Discrete Pulse Transform, when applied to* $f : \mathbb{Z}^d \to \mathbb{R}$, *also derives a related scale-space* $S_f^{LULU^C} = \{(n, P_n(f) - P_{n-1}(f)) : n \in \Lambda_0 = \{0, 1, 2, ..., N\}\} = \{(n, D_n(f)) : n \in \Lambda_0 = \{0, 1, 2, ..., N\}\}$ *which we call the* complimentary LULU scale-space.

Theorem 35 *If* $\mathcal{L}_f : \mathcal{A}(\Omega) \mapsto \mathbb{R}$ *satisfies the cascading property in Axiom 4 of Definition 30, then for every* $\lambda_1, \lambda_2 \in \Lambda, \lambda_1 < \lambda_2$,

$$\mathcal{L}_f(\lambda_1) = \mathcal{L}_g(\lambda_2) \Rightarrow \mathcal{L}_f(\lambda_2) = \mathcal{L}_g(\lambda_2).$$

Proof
By the cascading property we have

$$\begin{aligned} \mathcal{L}_f(\lambda_1) &= \mathcal{M}(\lambda_1, \lambda_2) \circ \mathcal{L}_f(\lambda_1) \\ &= \mathcal{M}(\lambda_1, \lambda_2) \circ \mathcal{L}_g(\lambda_2) \end{aligned}$$

■

Definition 36 *Given a measure of smoothness* S, *a function* $g \in \mathcal{A}(\Omega)$ *is an* event *of* $f \in \mathcal{A}(\Omega)$ *if*

$$S(f - g) + S(g) = S(f).$$

Definition 36 indicates that by removing g from f the smoothness has increased (or roughness has reduced) as a part of f has been removed.

Definition 37 *An event* $g \in \mathcal{A}(\Omega)$ *of* $f \in \mathcal{A}(\Omega)$ *is present at scale* λ *if* $\mathcal{L}_{f-g}(\lambda) \neq \mathcal{L}_f(\lambda)$.

Note also that if $g \in \mathcal{A}(\Omega)$ is an *event* of $f \in \mathcal{A}(\Omega)$ we have

$$S(\mathcal{L}_{f-g}(\lambda)) < S(\mathcal{L}_f(\lambda)).$$

Theorem 38 *For every* $\lambda_1, \lambda_2 \in \Lambda, \lambda_1 < \lambda_2$, *if* $g \in \mathcal{A}(\Omega)$ *is an event of* $f \in \mathcal{A}(\Omega)$, *then*

$$\mathcal{L}_f(\lambda_1) = \mathcal{L}_{f-g}(\lambda_1) \Rightarrow \mathcal{L}_f(\lambda_2) = \mathcal{L}_{f-g}(\lambda_2).$$

Proof
Follows by Theorem 35. ■

Theorem 38 shows that if an event is present at scale λ_1, the same event is present at scale $\lambda_2 > \lambda_1$.

The local maximum and minimum sets (see Definition 41) derived by the DPT are exactly such events.

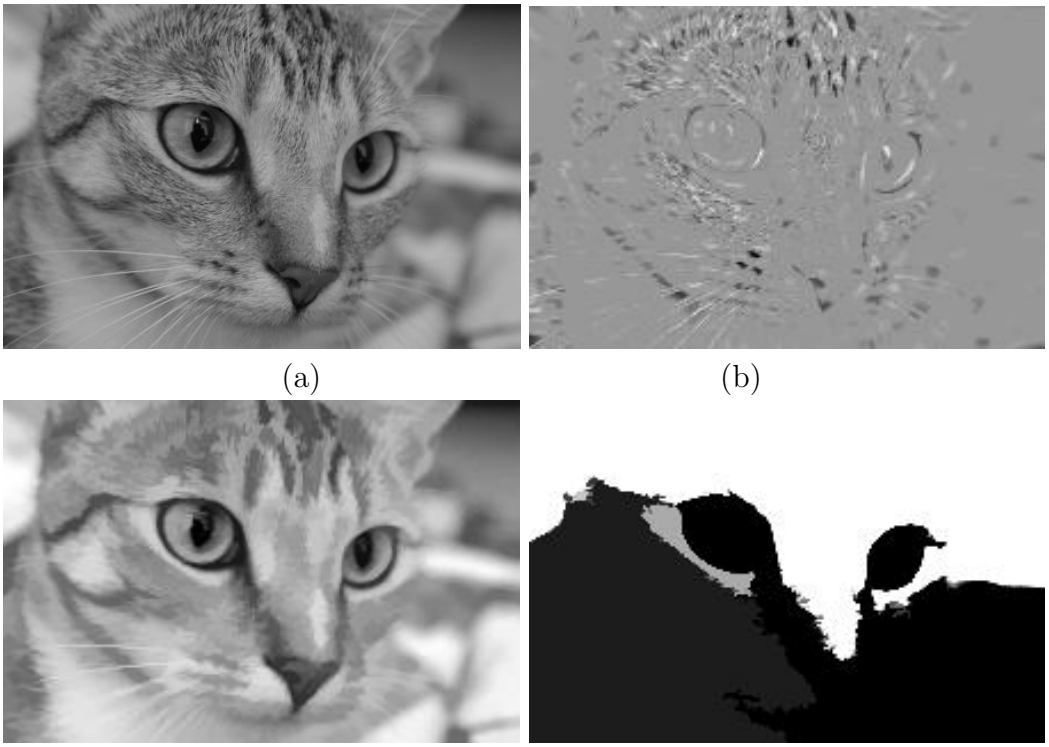
As mentioned before the Gaussian scale-space shifts and blurs edges through its scales and also does not correspond directly to object shapes at each scale [135]. The LULU scale-space does not suffer from this disadvantage. The LULU scale-space satisfies the axioms of the Gaussian scale-space as shown above but has the benefit of nonlinearity. This results in excellent shape and preservation properties, namely consistent separation, and total variation and shape preservation [8]. The Discrete Pulse Transform, $f = \sum_{n=1}^{N} D_n(f)$, forms a scale-space where the scaled image is $P_n(f)$ for discrete scales $n = 1, 2, 3, ..., N$. A second advantage of this LULU scale-space is then clear - it is already discrete and no approximations or sampling needs to be done, unlike with the Gaussian scale-space [12, 120, 109].

Often, a limited number of specific scales can sufficiently describe the important parts of an image, with discarded scales representing the background or noisy parts of the image. In addition, scales that repeat the representation of the same structures can be discarded or reduced, thereby reducing the amount of data but preserving the information contained in the image. Figure 4.3 gives an example of the break-down of an image into one possible LULU scale-space.

How do the individual pixel values change through the scale-space? We refer to this change over the scales as the *DPT pixel signatures*. Each pixel belongs to k pulses, $\phi_{n_1 s_{n_1}}, \phi_{n_2 s_{n_2}}, ..., \phi_{n_k s_{n_k}}$, at scales $\{n_1, n_2, ..., n_k\} \subseteq \{1, 2, 3, ..., N\}$. For each pixel x, we then have what we call a Discrete Pulse Vector (DPV) for a specified pixel $x \in \mathbb{Z}^2$,

$$\mathbf{p}_x = \begin{bmatrix} n_1 & n_2 & n_3 & \dots & n_k \\ \ell_1 & \ell_2 & \ell_3 & \dots & \ell_k \end{bmatrix}^T, \; x \in \mathbb{Z}^2 \tag{4.7}$$

where for each scale n_i, we have the corresponding relative luminosity ℓ_i of the pulse $\phi_{n_i s_{n_i}}$, that is, the height (positive) or depth (negative) of the local maximum or minimum set which pixel x belongs to at scale n_i. Figures 4.5 to



(a) (b)

(c) (d)

Figure 4.3: *A LULU scale-space for the Chelsea image (a) Original Image (b) Details - Scales* $n = 1$ *to* 35 *(c) Smoothed Image - Scales* $n = 36$ *to* 8000 *(d) Large Pulses - Scales* $n = 8001$ *to* $N = 33900$

Figure 4.4: *Original Canoeist image with the direction of the pixel indices indicated*

4.8 show the DPT pixel signatures of the canoeist, white water, dark water and 'normal' water areas of the image in Figure 4.4. The signatures indicated are similar for the same regions.

As a starting point in using the DPT for feature detection we investigate the DPV lengths. We detect the longest DPV's which represent pixels which are present over the most scales and reconstruct the image using only the discrete pulses that these pixels belong to. Figure 4.9 illustrates this idea. The method picks out the bottom left hand potato most likely because it has a background shadow as opposed to the rest of the potatoes. Figure 4.10 shows a similar result.

4.8 Scale-Space Applications

Scale-spaces have been used in a variety of applications namely image clustering and segmentation, deblurring and denoising of images, image enhancement, image compression, feature, corner and edge detection, as well as texture and shape analysis, to name a few.

In [118] the Gaussian scale-space is used to create a tree structure and then a stack approach used on this tree to segment the image. In [196] the hierarchical wavelet decomposition and Daubechieś four-tap filter are used to decompose the image into three detail images and a single approximate image. This is done recursively through the resulting pyramid to result in final improved segmentation via texture features. In [148] a hierarchical Markov Random Field (MRF) is used in segmenting high-resolution sonar images (in an unsupervised manner) using what they introduce as the scale casual

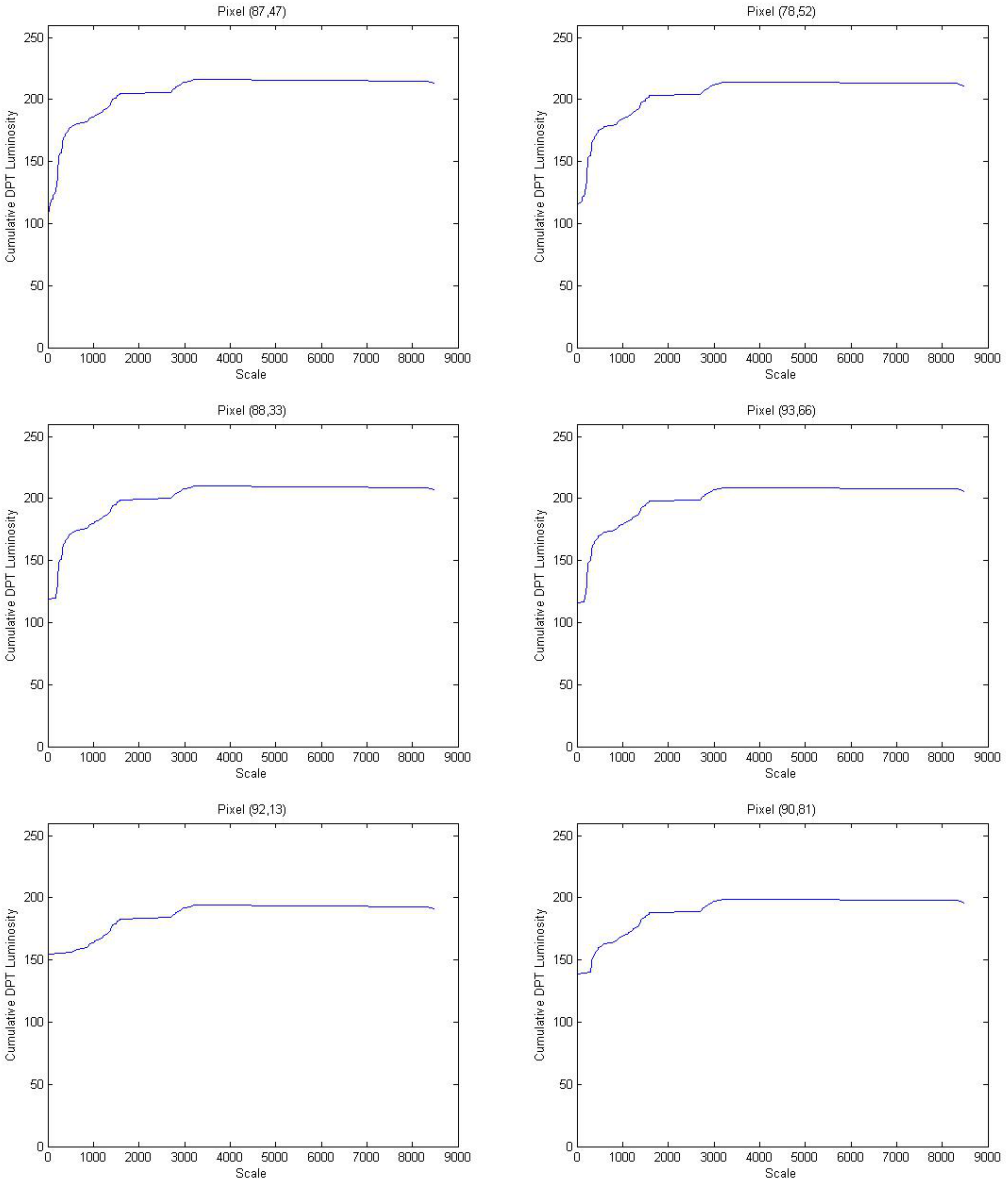


Figure 4.5: *DPT signatures of randomly selected pixels of the canoeist in the Canoeist Image*

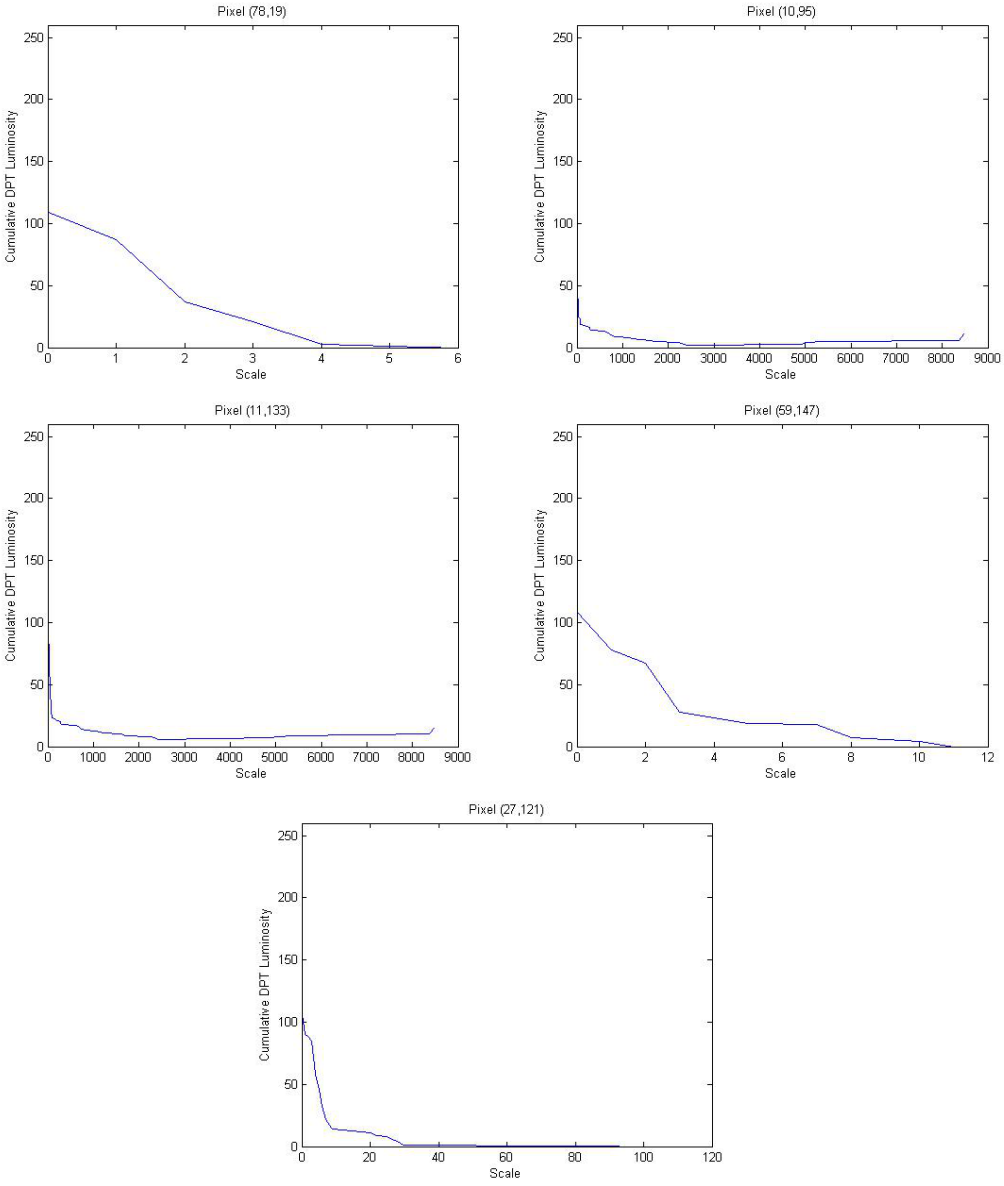


Figure 4.6: *DPT signatures of randomly selected pixels of the white water in the Canoeist Image*

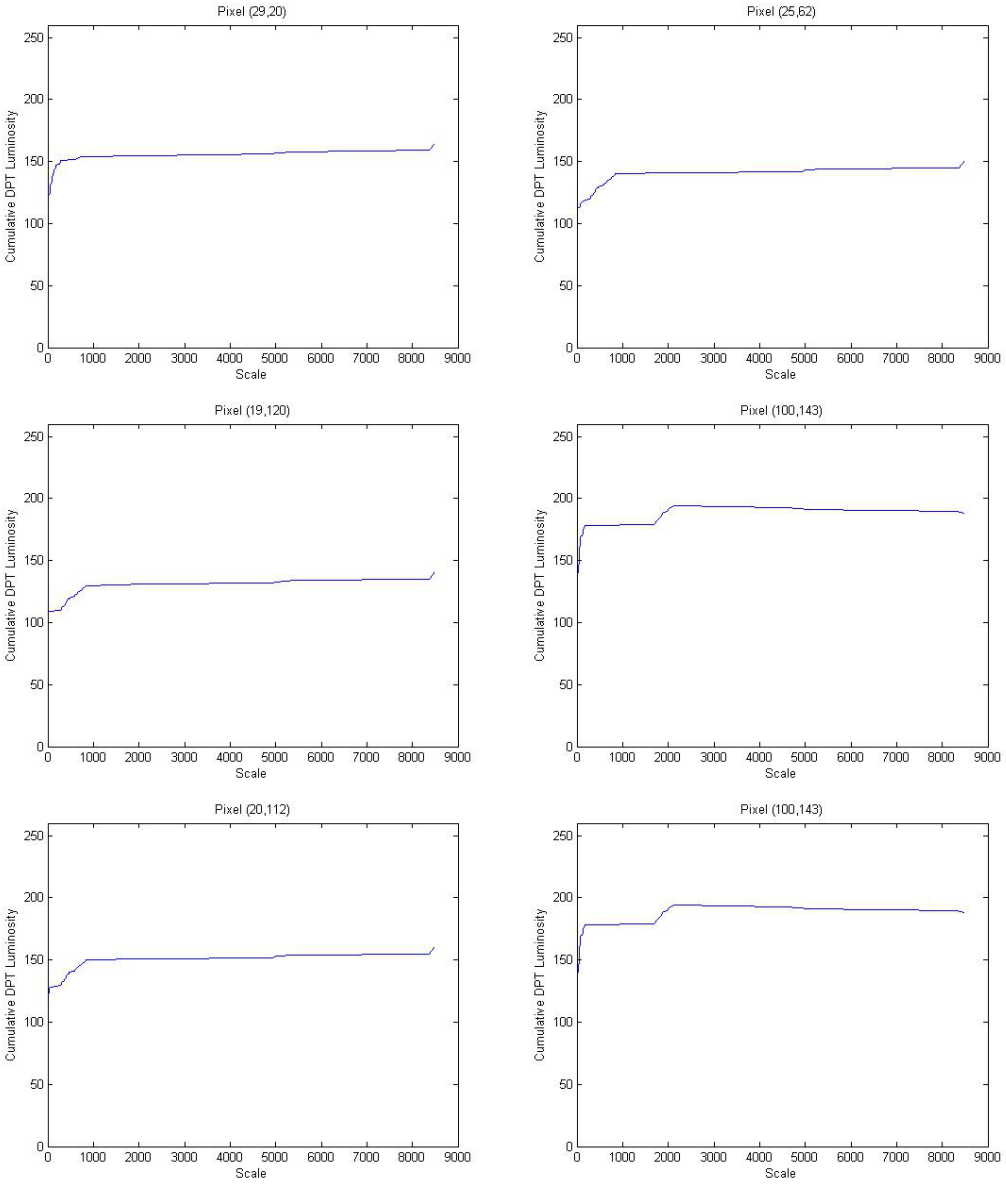


Figure 4.7: *DPT signatures of randomly selected pixels of the dark (shadowed) water in the Canoeist Image*

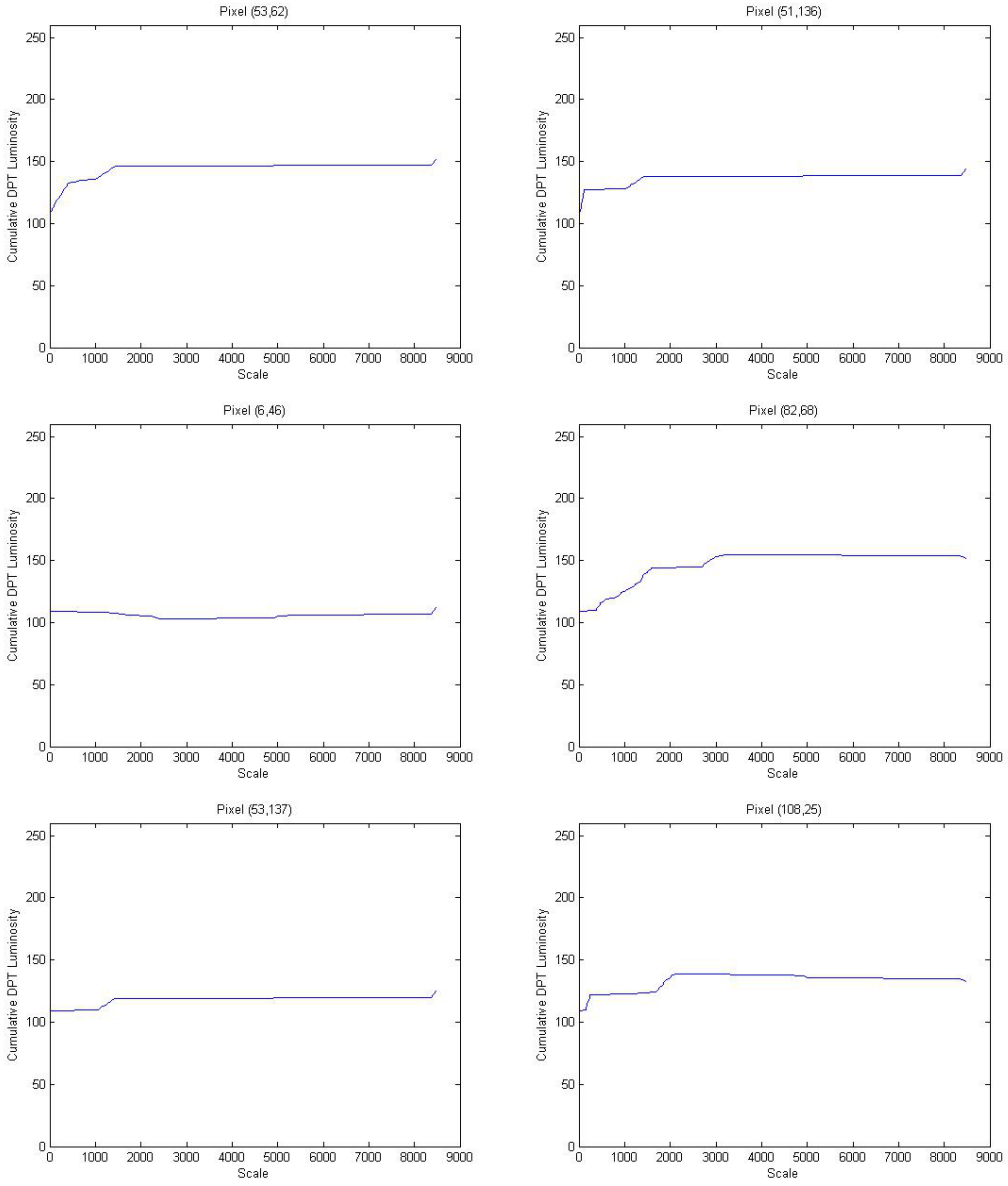
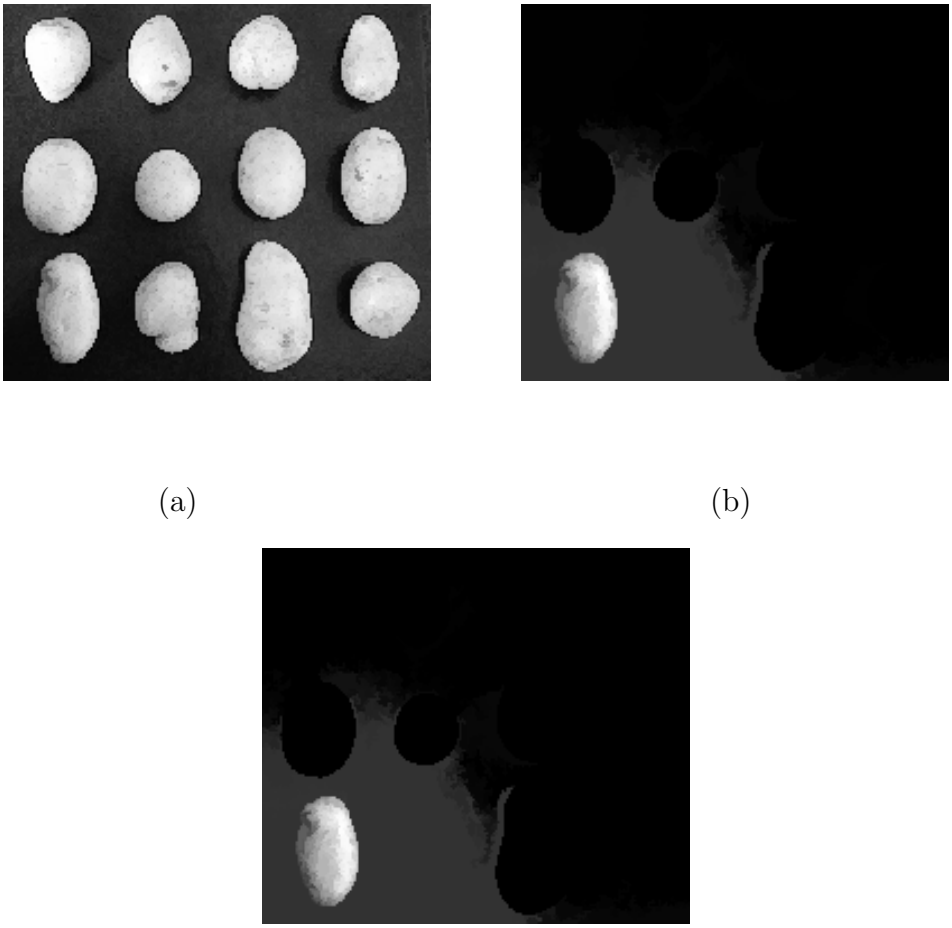


Figure 4.8: *DPT signatures of randomly selected pixels of the normal water in the Canoeist Image*



(a) (b)

(c)

Figure 4.9: *(a) Original (b) Ten largest DPV's (b) Largest DPV only*

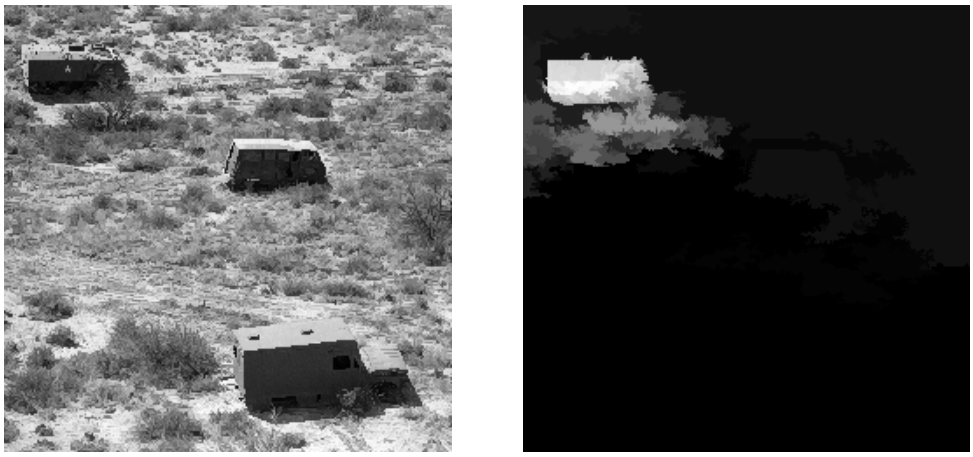


Figure 4.10: *(a) Original (b) 28 largest DPV's*

multigrid (SCM) algorithm. In [115] use of the Gaussian scale-space is justified as it simulates the action of the human visual system and a nested clustering dendrogram is produced such that data falling within the same region of the tree form a cluster in the segmentation. They also produce a non-nested hierarchical segmentation. In [146] a nonlinear scale-space via a general class of morphological levelings is presented and a brief description of how these levelings produce a segmentation is discussed. In [1] an improved segmentation is presented by using the morphological operators area open-close and close-open to produce a scale-space. In [111] use the Gaussian scale-space and its 'deep structure' to improve segmentation. In [171] a nonlinear scale-space is constructed via a diffusion equation, a tree is created and then unsupervised as well as supervised segmentation is presented via the edges through the scales of the scale-space. In [194] look at the spatial gradients between scales in the Gaussian scale-space and present a temporal segmentation of a sequence of images via the resulting scale-space tree. In [195] unsupervised discovery of valid clusters using statistics on the modes of the probability density function in the Gaussian scale space is shown.

In [192] multiscale total variation is introduced to improve their previous technique on textured regions for image recovery. In [51] a scale-adaption algorithm for reliable edge detection and blur estimation is presented. In [154] noise estimation is investigated via (i) multi-scale transforms, including wavelet transforms; (ii) a data structure termed the multiresolution support; and (iii) multiple scale significance testing. In [223, 222] an iterative varia-

tional decomposition via a minimizer functional is presented for deblurring, denoising and segmentation. In [247] a scale-space is developed via Markov Random Fields for the application of restoring degraded images.

In [19] an image compression scheme is introduced which involves a multiresolution decomposition derived from the wavelet transform. In [145] a cascade of compressions are produced via wavelet packets by coding the residual parts of each layer in a lossy manner which provides a sparse representation.

In [127] feature detection is determined via automatic scale selection in the Gaussian scale-space. In [228] a scale-space is created via pyramids of morphological operators and features are measured according to their persistence through the scales. In [149] multi-oriented, multi-scale Gabor filters are used to build a hierarchical model based on the visual cortex.

In [139] the raw primal sketch obtained with the Gaussian scale-space and its applicability for edge detection is presented.

In [177] the Gaussian scale-space is also made use of for corner detection. In [122] edge detection is investigated via automatic scale selection in the Gaussian scale-space. In [126] an edges strength is measured via the zero-crossings in the Gaussian scale-space and thereby enables edge detection. In [59] a thermodynamic model is employed for scale-space generation and significant edges (thin regions) are detected via this.

In [153] the curvature scale-space is presented (together with two additional versions of it) for shape representation at arbitrary scales and orientations. This provides insight into texture analysis. The author continues his work in [151, 152, 150] discussing shape matching similarity retrieval.

In [219] the author combines Shannons entropy and Witkin and Koenderinks scale-space to establish a precise connection between the heat equation and the thermodynamic entropy in Scale-Space. Experimentally the entropy function is used to study global textures.

Other applications of scale-spaces involve image fusion [225], image watermarking [173, 83], road extraction [141], astronomy [154], fingerprint enhancement [3], object tracking and recognition as well as image retrieval [105], surface editing in images [17] and palm print verification [119]. In [4] a multi-scale video analysis is described, an extension of their work for images, in which they introduce a new axiom namely that of Galilean invariance. This requires that the motion of the whole picture at constant velocity does not alter the analysis.

In [160] an imaged is decomposed into two images namely cartoon and the texture or noise and image deblurring denoising are presented as applications.

This is by no means an exhaustive list but simply an indication of the wide variety of applications in which scale-spaces are made use of.

4.8.1 Feature Detection in the Gaussian Scale-Space

One drawback of the Gaussian scale-space is its linearity. It removes small scale features (noise) very well but results in spatial distortions as scale increases, i.e. reduced sharpness of edges and shapes [129, 62]. To prevent this a nonlinear smoothing step is introduced in the literature, see [124, 170, 99, 63, 28, 230], and the LULU scale-space, obtained as the DPT, does the same (see [52] and [8] for the edge preserving properties of the DPT). Nonlinear filtering needs to be introduced into image analysis if realistic structures are the aim of the detection [123]. Koenderink and collaborators introduce the idea of using nonlinear, possibly, combinations of derivatives i.e. differential geometric descriptors, to introduce nonlinearity.

In [123, Chapter 6] a basic introduction into the use of the Gaussian scale-space and its scale-space derivatives for edge detection, junction (corner) detection and feature detection is presented. For edge detection the local directional derivatives are used to detect maximum gradient changes. Junction detection is obtained at high curvature combined with high gradient points . Feature detection is obtained by detecting zero-crossings and/or local extrema. Weickert et al [239] detect regions of interest as the stable stationary points in the Gaussian scale-space tree within a surrounding circular radius of appropriate radius.

4.8.2 Feature Detection in the LULU Scale-Space

With the availability of all the pulses of the Discrete Pulse Transform, the question arises as to how we can utilize all the obtained pulses to solve some of the problems encountered in image analysis? From the DPT, we no longer only have the original luminosity at each pixel, but instead have an otherwise invisible insight into the make-up and content of the image and the pixels within.

The additional image structure provided by the DPT provides improved fea-

ture detection over standard approaches using only luminosity values. This chapter will first look at the philosophy of feature detection in images and then introduce a number of techniques which utilize the DPT. The techniques investigated are rudimentary as we investigate the basis for using the DPT for feature point detection and feature detection. Advancements can be made once ability of the DPT in feature point detection as well as feature detection has been explored.

Philosophy of Feature Detection

In 1978 the ability of the human visual system (HVS) to discriminate an object in a random dot display was investigated by Barlow [13]. The aim was to determine estimates for the efficiency of the HVS to achieve this task, which Barlow refers to as absolute measures of sensory performance. He determined, albeit with a sample size of only 2, that the efficiency limit is 50% i.e. the HVS uses 50% of the data available for recognition tasks. In more obvious discrimination tasks this measured efficiency was less. Other experimenters determined similar results. Also in 1978, according to Barrow and Tenenbaum [15], the HVS easily characterizes a scene with respect to range, orientation, reflectance and incident illumination on first the first view. It contains cells which measure these individual characteristics and in a manner sums them to estimate the shape information [169]. Mishra and Jenkins [149] also designed feature extractors based on Gabor filters and motivate them with their link to detecting natural stimuli i.e. they are biologically inspired.

A feature extraction method needs to, in some manner, extract the signature of the objects in the scene. This should be done as accurately and uniquely as possible, as emphasized in [149], and the salient characteristics of the object should be measured. Salient is defined very nicely as *prominent* or *conspicuous* in the Oxford English Dictionary. A high profile paper in *Nature* [103], describes textons. These are local conspicuous features. The pre-attentive texture probing by the HVS uses these textons and first order moments for discrimination rather than higher order moments, that is, the simplest and most obvious is the most useful the descriptor. We can in fact go one step further than feature extraction and look into recognition of the feature detected. This then requires a type of classification based on the salient features we can extract from the object.

The aim of our feature detection using the DPT is to determine salient feature points of the image using the pulses in the DPT, as opposed to full features

extracted as objects (i.e. targets). Very importantly, we do not make any initial assumptions for the image regarding luminosity, amount of variation, size of objects present, texture etc., so that we can process any possible image. There does exist literature which assume a model for the image data, for example, a normal distribution [11], or a model for the way in which the image was obtained, for example when making use of the camera technique to remove illumination. In order to make our method applicable in all situations we shall ignore such ideas (although they are very useful when such assumptions are indeed true for a sample of images and thus improve the processing of the image). Of course, the interested reader could make improvements on our ideas if such assumptions are valid for their case.

How do we decide how salient a feature is? The most obvious is that large, high-contrast objects will naturally be more salient than small, low-contrast objects, in the absence of complicated backgrounds, but then at what size and contrast does the required saliency begin? In [107] this is determined by measuring the ability of the agent to draw a line around the target distinctively and they present a theory of optimal linear edge detection. According to Chi and Leung [35] humans recognize line drawings as quickly and almost as accurately as full detailed images. In addition they follow the five laws of Gestalt theory, which describe human perception of significant shape features, to set up good edge detectors. These five laws are *focal point*, that is select the top $a\%$ longest lines and arcs, *proximity*, *continuity*, *similarity* and *symmetry*, the latter four which choose the neighbours of the focal features concurrently according to these properties.

By using the edges or boundaries of an object we can also enter the field of shape analysis. Colour alone will not provide enough concrete data for detection as two vastly different objects present in an image may have the same luminosity. It is shape that represents the inherent structure of the image [50]. However, as mentioned in [69] by storing all the shape information we extract a huge amount of data from the image. There are measures available which can accurately describe a shape in a simple manner avoiding additional storage memory. This is also a very important strategy to be considered since the DPT produces a large number of pulses and thier analysis can require significant computational effort. The idea is to represent each object in the image with a feature vector, and not each pixel, thereby reducing the information that would need to be processed. Urdiales et al [227] describe some ideal properties of a feature vector, namely, uniqueness for each object, resistant to noise and as compact as possible for storage. Loncaric [130] gives a nice summary of shape analysis techniques.

Feature Extraction

In the absence of prior knowledge about the feature characteristics and size one has to keep every scale. Reflecting on how a human eye picks out features in an image the Human Vision System (HVS) model [105] provides some insight. It consists of a first stage, the *Pre-Attentive Stage*, in which the features are detected and then a second stage, the *Attentive Stage*, in which matching takes place between the detected features of the first stage and the rest of the image. It is clear that the HVS possesses a degree of saliency to detect the 'pop-out' features. We will show how the LULU operators can be used to detect these 'pop-out' features. Following the HVS model, the features are those areas in the image that are stable, that is, the areas that survive over a wide range of scales, [129, 122]. This is simple to apply to the LULU scale-space. Indeed, each pixel belongs to k pulses, $\phi_{n_1 s_{n_1}}, \phi_{n_2 s_{n_2}}, ..., \phi_{n_k s_{n_k}}$, at scales $\{n_1, n_2, ..., n_k\} \subseteq \{1, 2, 3, ..., N\}$. For each pixel x, we then have what we call a Discrete Pulse Vector (DPV) for a specified pixel $x \in \mathbb{Z}^2$,

$$\mathbf{p}_x = \begin{bmatrix} n_1 & n_2 & n_3 & \dots & n_k \\ \ell_1 & \ell_2 & \ell_3 & \dots & \ell_k \end{bmatrix}^T, \; x \in \mathbb{Z}^2 \tag{4.8}$$

where for each scale n_i, we have the corresponding relative luminosity ℓ_i of the pulse $\phi_{n_i s_{n_i}}$, that is, the height (positive) or depth (negative) of the local maximum or minimum set which pixel x belongs to at scale n_i. The simplest and most obvious way of using these DPV's for feature detection is by keeping only those pixels belonging to DPV's that contain a large number of scales, i.e. large values of k. This is illustrated in Figure 4.11. Whiter values (higher luminosity) indicate larger values of k. In the last image only the top 20% proportion of the largest values of k and their respective pixels are kept. Notice how the front of the tank is a strong feature. We refer to the value k as the *impulse strength*. Van der Walt refers to this as the *strength of the pixel* [231].

An alternative is to consider the ranges $n_k - n_1$, referred to as the *scale-space lifetime* at the pixel [122], but this method does not differentiate between features as effectively as the first. Compare Figure 4.11 with Figure 4.12. We clearly see this measure does not pick out dominant features as well. For this method one needs to probably do some removal of outliers and cleaning of the data first. It still picks out the tank as a feature (seen in white) but the background also gets picked out and remains even if we threshold. This

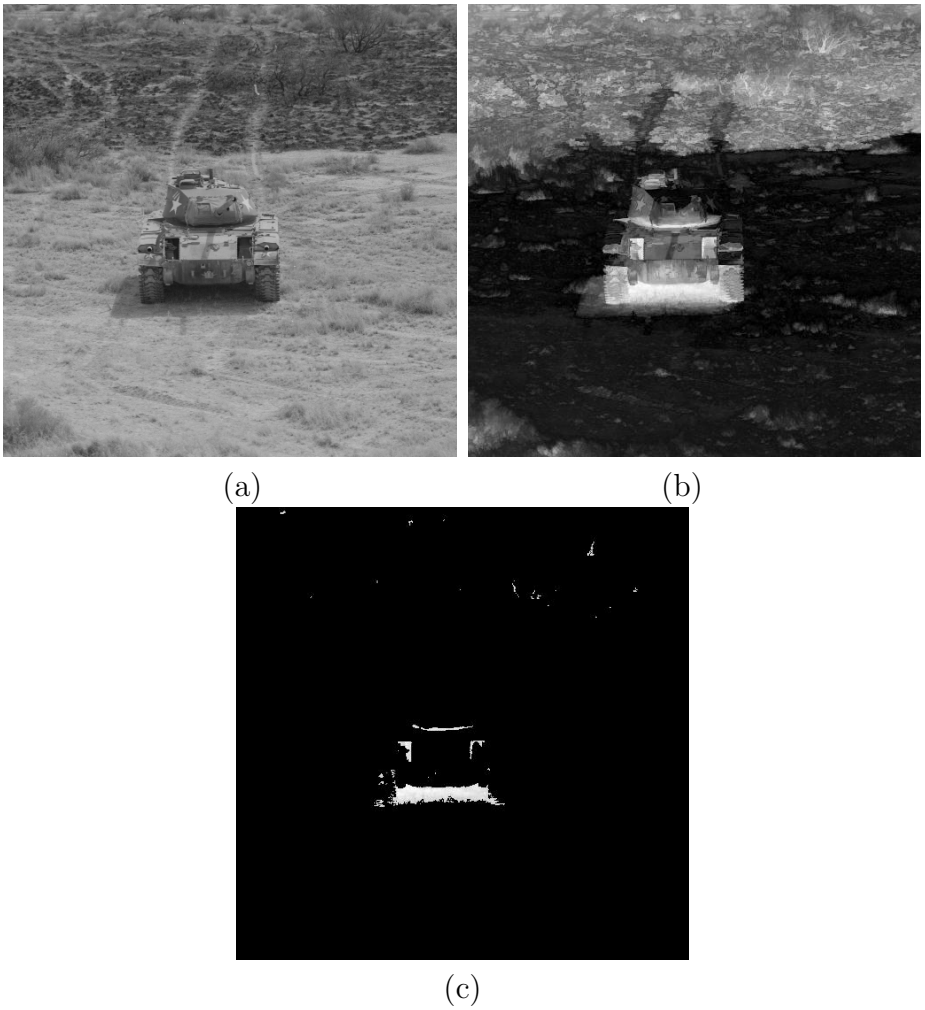


Figure 4.11: *(a) The original tank image with (b) its impulse strength shown, as well as (c) only the top* 20% *largest impulse strength pixels*

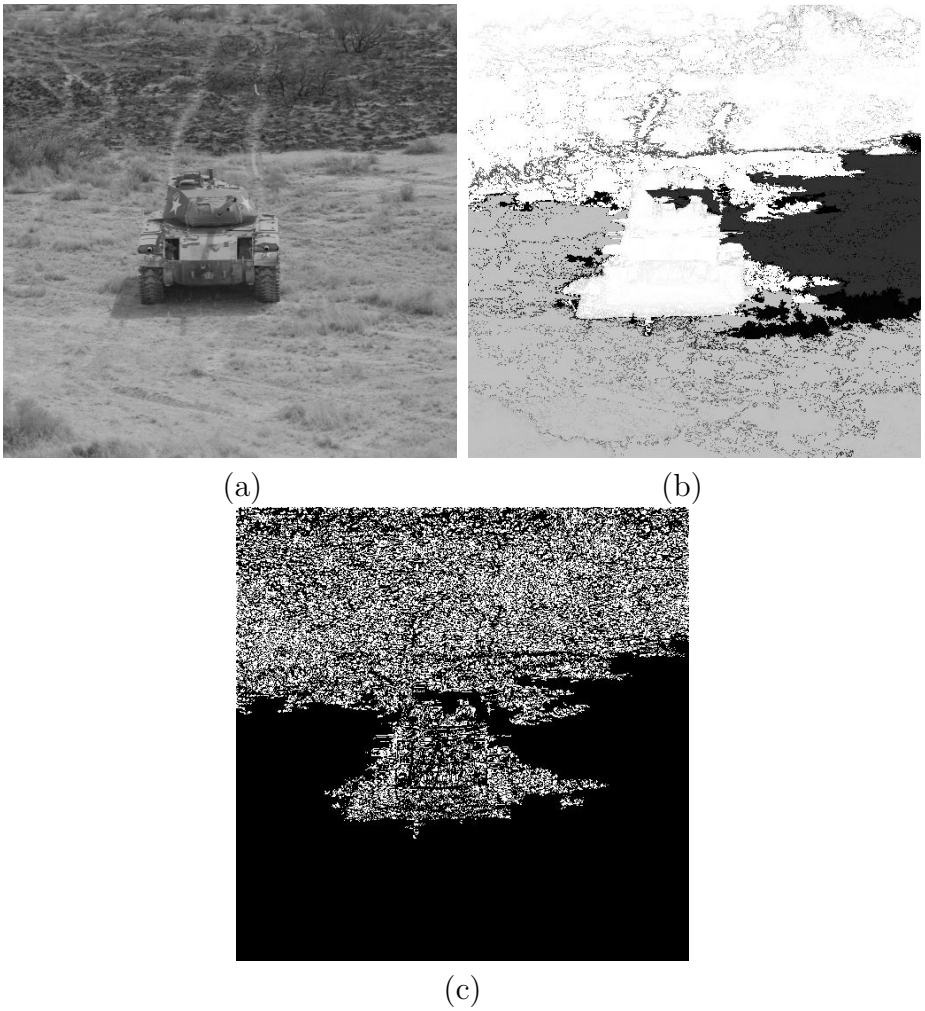


Figure 4.12: *(a) The original tank image with (b) its scale-space lifetimes shown, and (c) only the top* 0.01% *largest scale-space lifetime pixels*

can be understood logically, however, since a discrete pulse vector may be for example only of length 2 and have only a small scale and a very large scale, giving a large value for $n_k - n_1$. The pixel however does not exist over a large range of scales and should not be classified as such. The scale-space life-time may however provide an indication of whether a pixel is noise, texture, small detail, large detail etc.

As mentioned in [122] a two stage approach may be better, thus we present a method in which first the feature are detected via impulse strength mentioned above and then fine-tuned using finer scale data and shape descriptors. We present three examples to illustrate this idea. Further research is currently

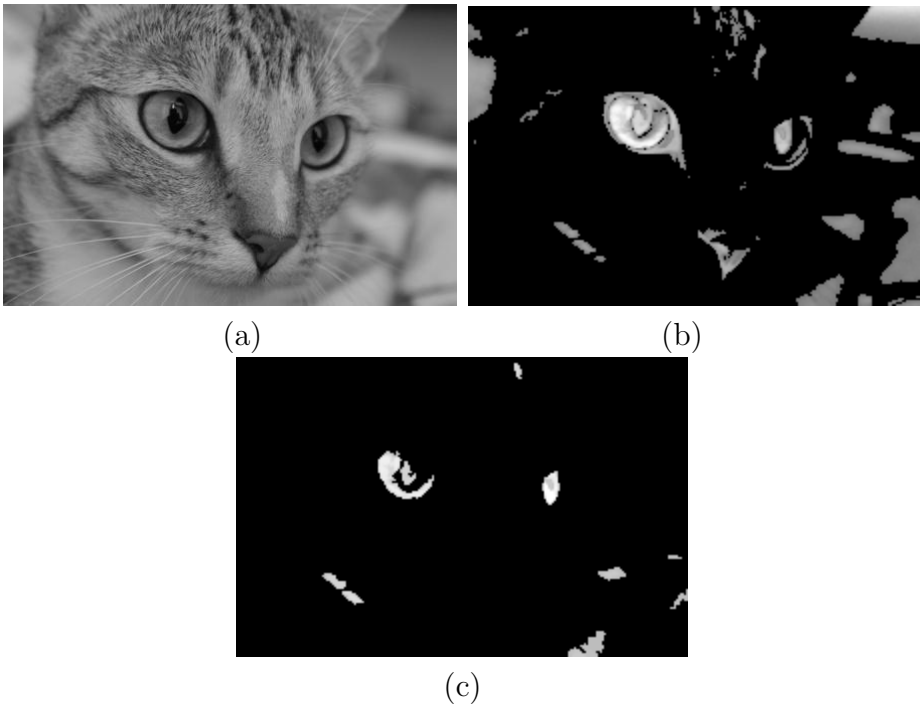


Figure 4.13: *Pulse strength illustrated on (a). (b) Only pixels included in at least 59 out of the 126 possible pulse scales are shown. (c) The circularity of the pulses used in (b) is restricted to between 0.3 and 0.6 to extract the eyes of the cat.*

being conducted to perfect this technique.

In Figure 4.13, we keep only the pixels which have discrete pulse vectors with at least 59 scales out of the maximum of 126 over all the pixels. We can see that the cat's two eyes and nose are picked out as features. We also see that some large background pulses are detected as features. These large noise pulses can be filtered out with a circularity shape descriptor. A circularity value close to 1 then indicates higher circularity than a value closer to 0.

In Figure 4.14, we keep only the pixels that have discrete pulse vectors with at least 65 scales out of a maximum of 105 over all the pixels. The three vehicles are detected as features. In Figure 4.15, we first remove the glint on the surface of the ocean by limiting the luminosity of the individual pulses. The third image in Figure 4.15 indicates the impulse strength of the image with the glint removed. The two main features in this image are the yacht and surprisingly the atmospheric mist effect on the land sea border. Atmospheric conditions often affect feature detection in marine images. In addition, when

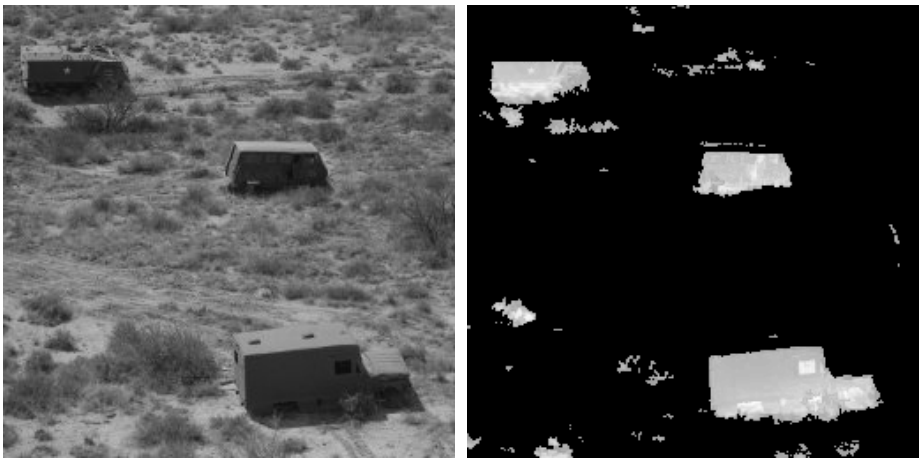


Figure 4.14: *Impulse strengths illustrated on* Trucks & APCs.

we talk about features we refer to those 'objects' which 'pop-out' first, and indeed when looking at the image in Figure 4.15 the yacht and the mist stand out first. Notice also though that the small boat in the background is also picked out via the impulse strength, even though it is very dark and almost camouflaged into the water. We can filter out the effect of the mist, and other effects, by using only pulses with specific areas, see the fourth image in Figure 4.15. The small boat is picked out in a similar manner.

These examples give an overview of the capability of the DPT for feature detection.

4.8.3 Segmentation in the LULU Scale-Space

Segmentation is the process of partitioning an image or signal into segments which provide a simpler representation more indicative of the image content with respect to visual characteristics. Serra provides a formal definition in terms of partitions and connectivity in [213]. It is immediately obvious that different segmentations could be obtained by using different measures for the similarity of image content. An obtained segmentation may be over-segmented meaning there exist some pairs of regions for which the between-region variation is small compared to the within-region variation, so that there are too many regions in the segmentation, [57]. In [163] a connection is presented which can be used in place of the usual image connectivity to avoid over-segmentation. An image may also be under-segmented meaning there

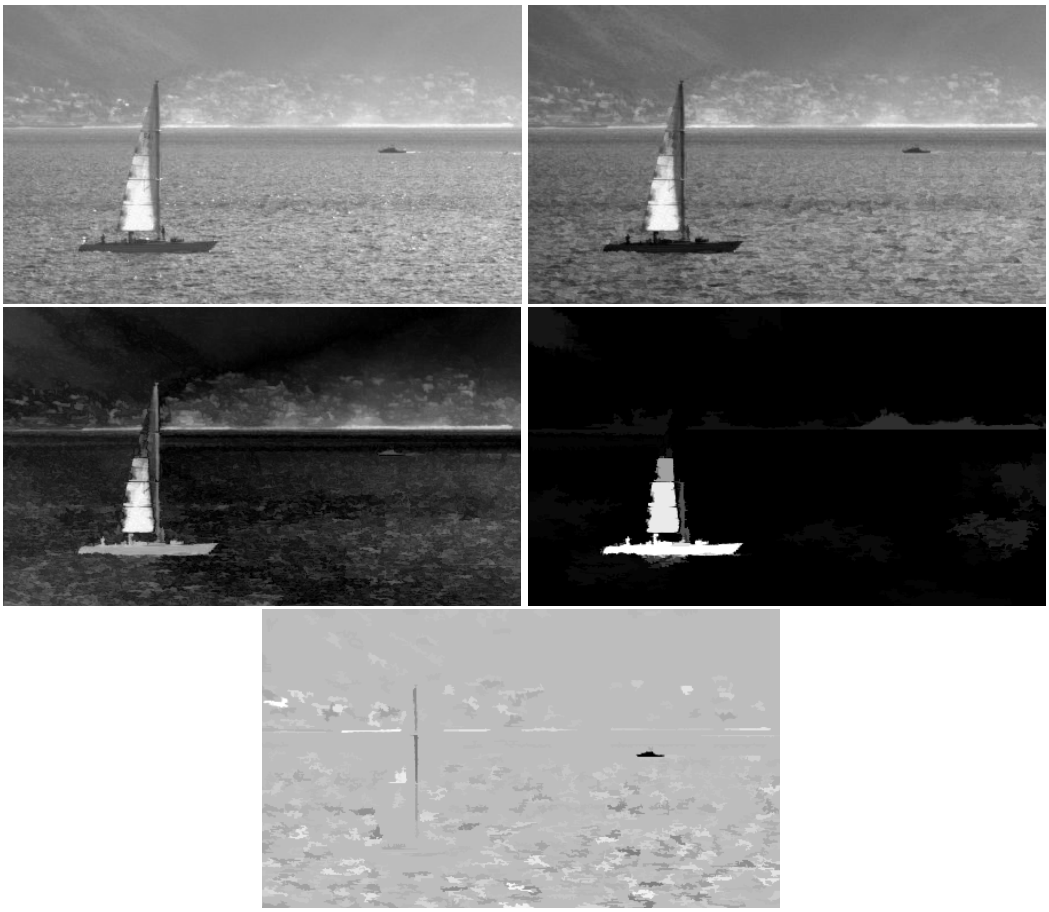


Figure 4.15: *Impulse strengths illustrated on* yacht.

exists a way to further segment regions without causing over-segmentation so that there are too few regions in the segmentation [57]. Felzenszwalb and Huttenlocher [57] and Hoover et al [82] provide examples of graph-theoretical and set-theoretic definitions for over- and under-segmentation respectively. Zhang discusses techniques to measure the quality of a segmentation, namely analytical, which involves looking at the actual algorithm and its properties, most importantly convergence properties; empirical goodness, which is based on human perception of a good segmentation; and empirical discrepancy, which involves a comparison with the ground truth segmentation, if available [253].

Algorithms for segmentation may take into account a priori information about an image. Methods like this, when the features in the image are known, are called supervised segmentation . Methods may also be semi-supervised in which case features are only partially known. Unsupervised methods assume no knowledge of the image features and learn as the algorithm proceeds. Song and Fan [218] present a study on the different techniques based on the availability of image features. The number of segments an algorithm should aim for is also a problem which has been given attention. If this is known before hand it can be specified. Other methods determine this as the algorithm proceeds. For example, Nakamura and Kehtarnavaz [155] provide a method to determine the appropriate number of clusters by making use of scale-space theory in which a prominent data structure is one which survives over many scales and Sakai and Imiya [195] use the modes of a probability density function obtained via the Gaussian scale-space for cluster discovery.

As discussed in detail in Chapter 4, the human vision system has a big effect on the philosophy of imaging techniques, and this is true for segmentation as well. Zahn segments into Gestalt clusters which are those perceived by humans [251]. Ramos et al base a segmentation into strong edges, smooth regions and textured regions on psychophysical studies [176]. Leung et al present a clustering by using scale-space's to simulate the human visual system [115].

Segmentation using connected operators has proved very effective. Salembier and Serra [200] argue for the use of filters by reconstruction since they simplify the image while preserving contours and are thus good for noise cancelation and improved segmentation. In [215] the same authors use pyramids of nested flat zones based on connected operators. This also provides good segmentation since simplified into flat zones and preserves contour information. Soille [217] goes further and deals with a constrained connectivity for

which 2 pixels are connected if they satisfy a series of constraints in terms of simple measures. He uses this connectivity for segmentation.

Using the pulses of the DPT we obtain improved segmentation. Each pixel in the image belongs to a number of pulses in the DPT but not at every scale. We associate a *Discrete Pulse Vector* (DPV) with each pixel, namely

$$DPV(x) = \begin{pmatrix} s_1^{(x)} & s_2^{(x)} & ... & s_m^{(x)} \\ \ell_1^{(x)} & \ell_2^{(x)} & ... & \ell_m^{(x)} \end{pmatrix},$$

where the $s_i^{(x)}$'s for $i = 1, 2, ..., m$ are the scales at which pixel x appears in a pulse and the the relative luminosities $\ell_i^{(x)}$'s for $i = 1, 2, ..., m$ are the respective heights or depths of the pulse at that scale containing x. Various pixels may be present in a large number of scales resulting in very large DPV's as well as DPV's of different lengths so the DPV's cannot be clustered directly. This information needs to be summarized into only a few values in order for each pixel to be clustered using the algorithm. We investigated using the following possible summarizing measures,

- $\sum_{i=1}^{m} |\ell_i|$
- $\sum_{i=1}^{m} |\ell_i| s_i$
- $\sum_{i=1}^{m} \ell_i s_i$
- $\sum_{i=1}^{m} \ell_i \sqrt{s_i}$
- $\sum_{i=1}^{m} (\ell_i)^2 s_i$.

The investigations indicate that $\sum_{i=1}^{m} |\ell_i|$ performs best in representing the content of the image obtained from the DPT and we use this measure throughout. The measure can in addition be broken into bands

$$\sum_{i=1}^{m_1} |\ell_i|, \sum_{i=m_1+1}^{m_2} |\ell_i|, ..., \sum_{i=m_n+1}^{m} |\ell_i| \tag{4.9}$$

and a vector clustering algorithm applied. As long as the number of bands is not too large this is fairly simple and provides better segmentations.

We make use of the FCM algorithm for initial illustrations. We present some examples in Figures 4.16 to 4.20. FCM is an alternative to the standard k-means algorithm and incorporates a degree of fuzziness with respect to

the cluster assignments, as opposed to the hard clustering of the k-means algorithm where each observation can belong to only one cluster. Duda and Canty [48] compare a number of algorithms and conclude that fuzzy association works the best. Each observation x_p is assigned a coefficient $w_i(x_p)$ representing the degree of association of x_p with cluster i such that $\sum_{i=1}^{c} w_i(x_p) = 1$ for each x_p. A larger coefficient indicates a better strength of association with that respective cluster. The centers are calculated as

$$\mu_i = \frac{\sum_{x_p} w_i(x_p)^m x_p}{\sum_{x_p} w_i(x_p)^m}$$

where m is the fuzzy exponent (usually taken as 2) and the coefficients updated as the inverse distance from the observation to the cluster

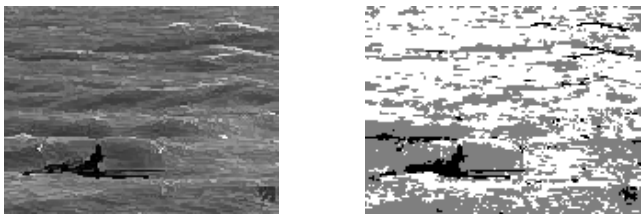
$$w_i(x_p) = \left(\sum_{j=1}^{c} \left(\frac{\|\mu_i - x_p\|}{\|\mu_j - x_p\|} \right)^{2/(m-1)} \right)^{-1}.$$

Convergence of the fuzzy c-means (FCM) algorithm is obtained when the coefficients no longer change significantly. The final segmentation is obtained by assigning observations to the cluster i for which w_i is the largest of the coefficients for that observation. The FCM algorithm results in similar poor segmentation sometimes. Gath and Geva [70] provide an unsupervised FCM algorithm which determines the number of clusters as it proceeds. Xie and Beni [249] introduce a validity measure for the FCM clusters. Krishnapuram and Keller [110] compare fuzzy and hard k-means with possibilistic clustering since the former two encounter trouble in noisy environments. Possibilistic clustering softens the requirements on the fuzzy coefficients such that $\sum_{i=1}^{k} w_i(x_p) \leq 1$ for each x_p. Pal et al [165] also include a possibilistic element to the algorithm to improve its effect on noise. Hammah and Currah [75] look at using different distance measures and how they affect the algorithm. They also introduce a new measure based on the Kent probability distribution.

In Figure 4.16 the improved segmentation using the LULU scale-space is shown. In Figure 4.17 the same is illustrated on the sharpened image (see Section 5.2) giving similar results, but in fact the segmentation appears worse. This can be attributed to the low resolution of the image. We include it none-the-less as it does provide insight into the cluster scale distributions. Figure 4.18 shows the distributions of the scales represented by the three clusters of Figure 4.16(b). Although similar there are distinct difference too. The black cluster, for example, represents more smaller scales than larger scales, as opposed to the white cluster.

(a) (b) (c)

Figure 4.16: *(a) Original Image, (b) Clustering the DPT into 3 clusters, (c) Ordinary FCM with 3 clusters*



(a) (b)

Figure 4.17: *(a) Sharpened Original Image, (b) Clustering the DPT into 3 clusters*

The fact that the DPT provides us with pulses of every size also allows us to remove certain scales before segmentation. Figure 4.19 illustrates this idea on the previous example. Only pulses larger then 150 are used in the segmentation. The result is a very sound segmentation. The scale distributions amongst the 3 clusters are given in Figure 4.20 yielding similar results.

The ICM clustering algorithm presented in [46] is effective yet simple enough to illustrate improved segmentation as well. The ICM algorithm follows. Notice that k-means is used as an initial step for the algorithm providing even better segmentation.

ITERATED CONDITIONAL MODES ALGORITHM

For a segmentation of an image I with N pixels represented by (i, j), and given feature vectors f_{ij} for each pixel, into K clusters $C_1^{(\alpha)}, C_2^{(\alpha)}, ..., C_K^{(\alpha)}$ where α is the number of iterations the steps proceed as follows:

1. Use the k-means algorithm to obtain initial cluster mean vectors $\mu_k^{(0)}$ for clusters $k = 1, 2, 3, ..., K$.

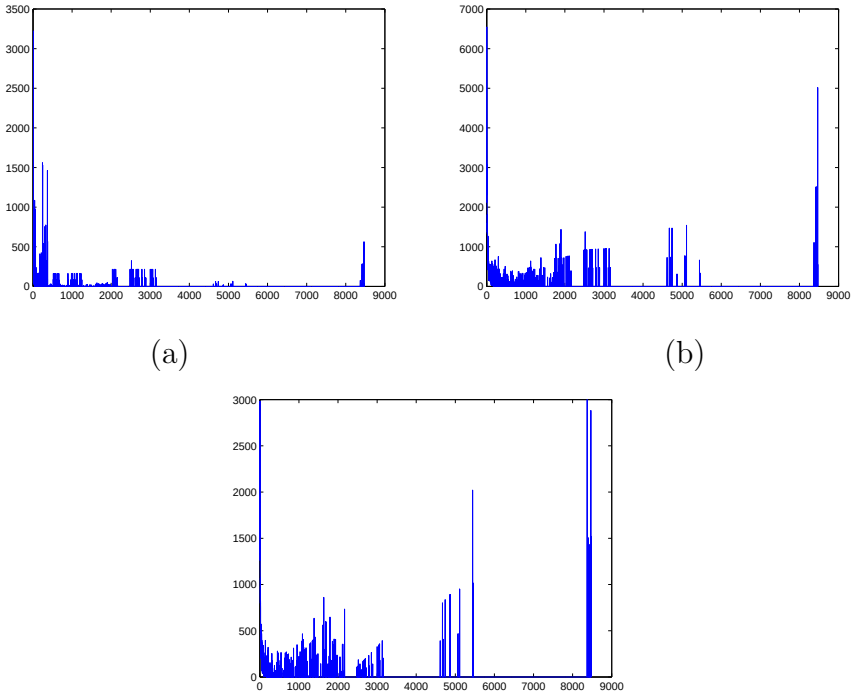


Figure 4.18: *Scale distributions within each cluster of Figure 4.16(b) (a) Black Cluster, (b) Grey Cluster, (c) White Cluster*

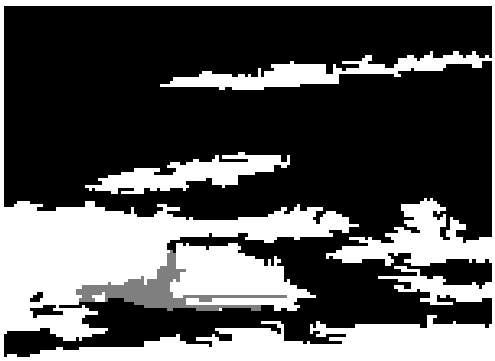
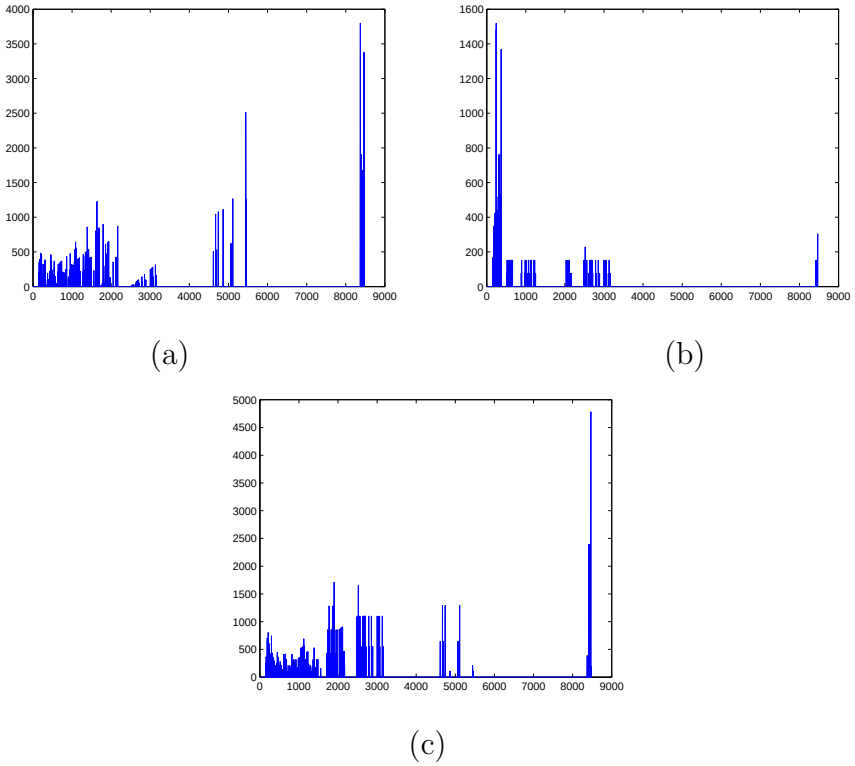


Figure 4.19: *Clustering of the sharp DPT into 3 clusters using only pulses larger than 150 i.e.* $f_{new} = \sum_{n=150}^{N} D_n(f)$



(a) (b)

(c)

Figure 4.20: *Scale distributions within each cluster of Figure 4.19 (a) Black Cluster, (b) Grey Cluster, (c) White Cluster*

2. Assign pixel (i, j) to cluster k for which the minimum of

$$\left(f_{ij} - \mu_k^{(\alpha)}\right)^T \left(f_{ij} - \mu_k^{(\alpha)}\right) - \beta\nu^{(\alpha)} N_{ij}^{(\alpha)}(k)$$

is obtained, where

- β is a spatial penalization parameter (suggested as 1.5 in [46]),
- $\nu^{(\alpha)} = \frac{1}{N}\sum_{k=1}^{N}\sum_{(i,j)\in C_k^{(\alpha)}} \left(f_{ij} - \mu_k^{(\alpha)}\right)^T \left(f_{ij} - \mu_k^{(\alpha)}\right)$ is the within cluster variance, and
- $N_{ij}^{(\alpha)}(k)$ is the number of neighbours of pixel (i, j) currently classified in cluster k at iteration α.

3. Recalculate the cluster mean vectors

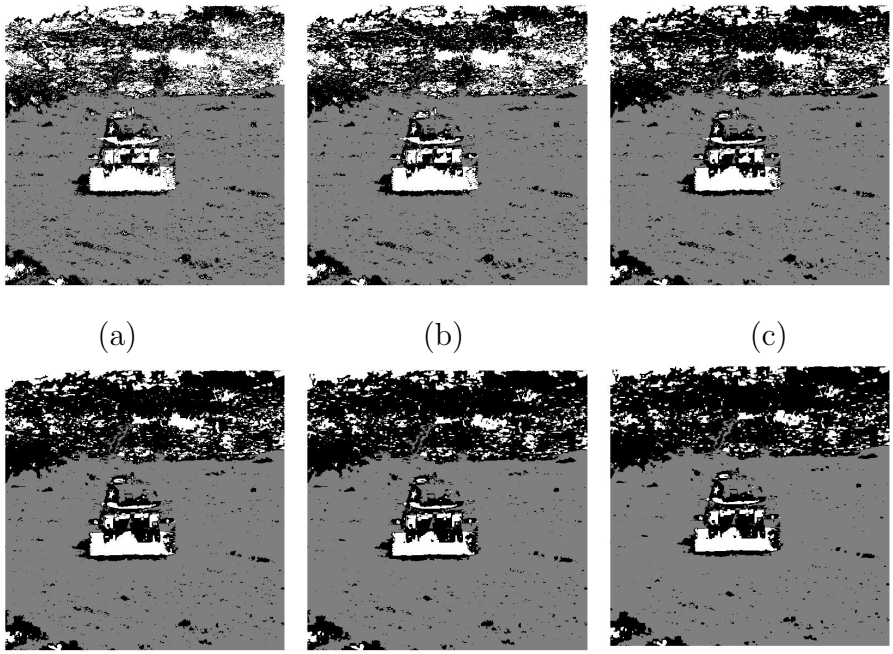
$$\mu_k^{(\alpha)} = \frac{1}{N_k^{(\alpha)}} \sum_{(i,j)\in C_k^{(\alpha)}} f_{ij}.$$

4. Repeat steps 2 and 3 until convergence (no change).

We illustrate the effect of β in Figure 4.21. Notice how the regions in the image are more smoothed with less detail as β increases.

We repeat the ICM segmentation on the image used to illustrate the k-means algorithm in Figures 4.22 to 4.24 . Figure 4.22 shows the ICM algorithm applied to the original image without the use of the DPT. Notice the improvement over the k-means results already in the 3 cluster segmentation. In Figure 4.23 the segmentation is done with the DPT. There doesn't seem to be a huge improvement and in fact the segmentation requires 5 clusters now to pick up the canoeist effectively. However, in Figure 4.24 the segmentation is done again using the DPT but only pulses larger than 100, as was discussed in Chapter 4.8.2 as significant structures are very unlikely to be this small (depending on the total image size of course). The canoeist is picked out in the 3-, 4- and 5-cluster segmentation in this case. It is not surprising the 2-cluster segmentation cannot pick the canoeist out as there are clearly three patterns in the image - the dark water, white water and the canoeist, thus the canoeist will end up being classified with one of the water patterns.

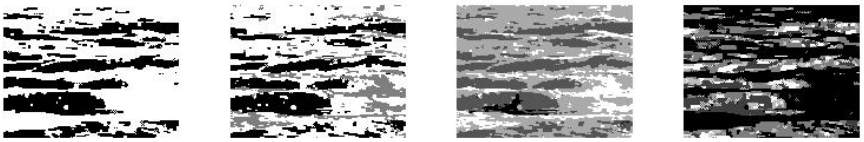
We now look at the Tank image introduced in Figure 4.21. Figure 4.25 presents the ICM segmentation of the Tank image without using the DPT.



(a) (b) (c)

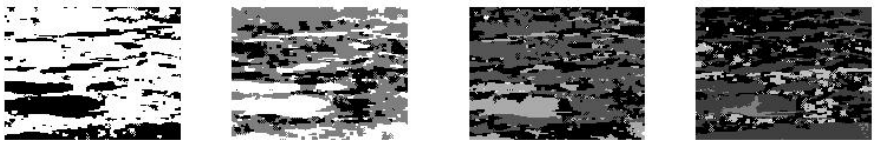
(d) (e) (f)

Figure 4.21: *The effect of parameter β in the ICM algorithm illustrated on the Tank image clustered into 3 clusters (a) $\beta = 0.1$ (b) $\beta = 0.5$ (c) $\beta = 1$ (d) $\beta = 1.5$ (e) $\beta = 2$ (f) $\beta = 2.5$*



(a) (b) (c) (d)

Figure 4.22: *ICM segmentation illustrated on the Canoeist for (a) 2 (b) 3 (c) 4 (d) 5, clusters*



(a) (b) (c) (d)

Figure 4.23: *ICM segmentation of the Canoeist image using the DPT into (a) 2 (b) 3 (c) 4 (d) 5, clusters*

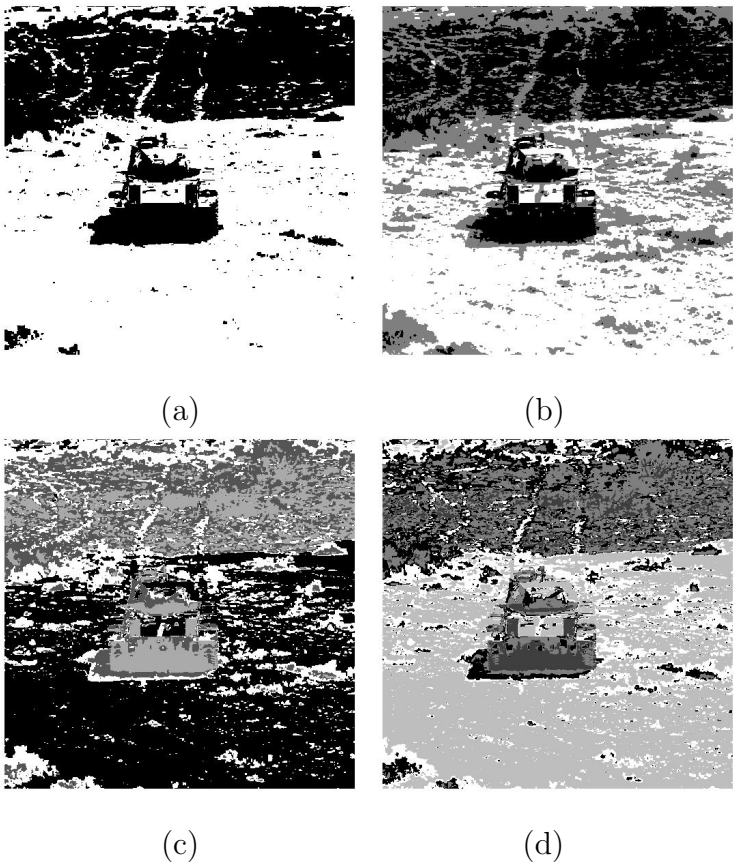
(a) (b) (c) (d)

Figure 4.24: *ICM segmentation of the Canoeist image using the DPT and only pulses 100 and larger into (a) 2 (b) 3 (c) 4 (d) 5, clusters*

Notice has 'messy' the segmentations are - the clusters are not easily discernable. Figure 4.26 shows the improved segmentation using the DPT. The segmentations are more clear.

Since the DPT provides us with all the scale information and the ICM algorithm can be vectorized, further improved segmentation may be obtained by separating the $|\ell_i|$ into bands indicated in (4.9). Figure 4.27 shows this method by separating the number of scales in half, the lower half representing the smaller scales and the upper half the larger scales. Notice that in the vector segmentation on (a) already shows an improvement over Figure 4.26(b). In Figure 4.27(b) the algorithm does not converge as there are not significantly different patterns in the information provided by the lower scales. Figure 4.27(c) also provides better segmentation - the background grass segments more consistently than before. Figure 4.28 shows the segmentation by separating the scales into three bands. By applying the ICM algorithm to the lower and middle band individually doesn't result in convergence and are thus not included.

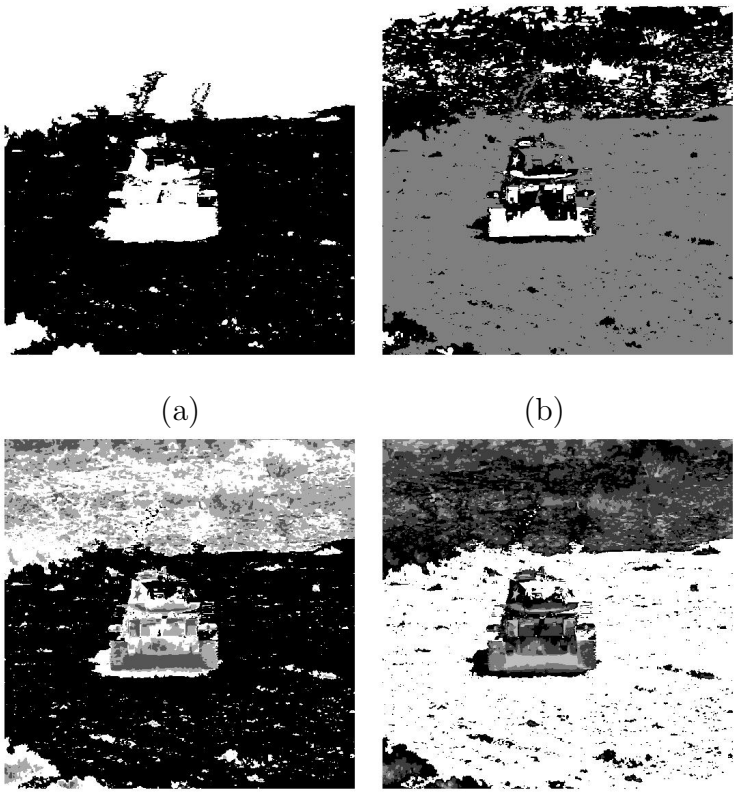
By using the total variation spectrum we can improve the grouping used above. Figure 4.29 shows the variation spectrum [52] for the *Tank* image. There seem to be five distinct bands of total variation, namely, 1 - 30000,



(a) (b)

(c) (d)

Figure 4.25: *ICM segmentation of the Tank image into (a) 2 (b) 3 (c) 4 (d) 5, clusters*



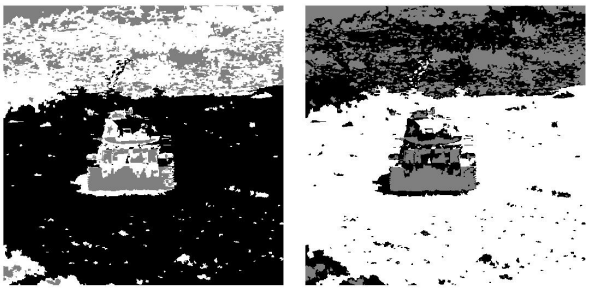
(a) (b)

(c) (d)

Figure 4.26: *ICM segmentation using the DPT of the Tank image into (a) 2 (b) 3 (c) 4 (d) 5, clusters*

(a) (b) (c)

Figure 4.27: *ICM segmentation of the Tank image into three clusters using the DPT separated into two bands (a) both bands clusters (b) lower band clustered (c) upper band clustered*



(a) (b)

Figure 4.28: *ICM segmentation of the Tank image into three clusters using the DPT separated into three bands (a) both bands clustered (b) upper band clustered*

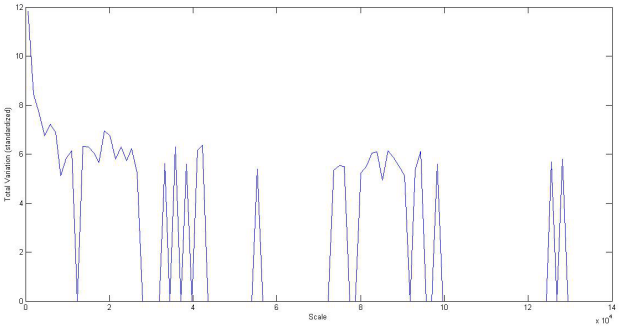


Figure 4.29: *Total Variation Spectrum of the Tank Image*

30001 - 50000, 50001 - 70000, 70001 - 120000, 120001 - 130139. The result of vectorized ICM segmentation using the total variation spectrum is shown in Figure 4.30. The individual segmentations of bands 1 - 30000, 30001 - 50000 and 50001 - 70000 do not converge illustrating the information within each of these bands has low variation. Figure 4.30(a) and (e) present the best segmentations by picking out the two different background grass shades, some significant features in the grass, as well as the tank with its different features.

In Chapter 4.8.2 it was discussed that pulses of size larger than 100 should be used for feature detection as significant structures are very likely to be smaller (depending on the total image size of course). Incorporating this into segmentation gives the results in Figure 4.31. We combine this idea with the total variation spectrum and investigate bands 100 - 30000, 30001 - 50000, 50001 - 70000, 70001 - 120000, 120001 - 130139. The result is shown in Figure 4.31(e) - notice it is very similar to (b).

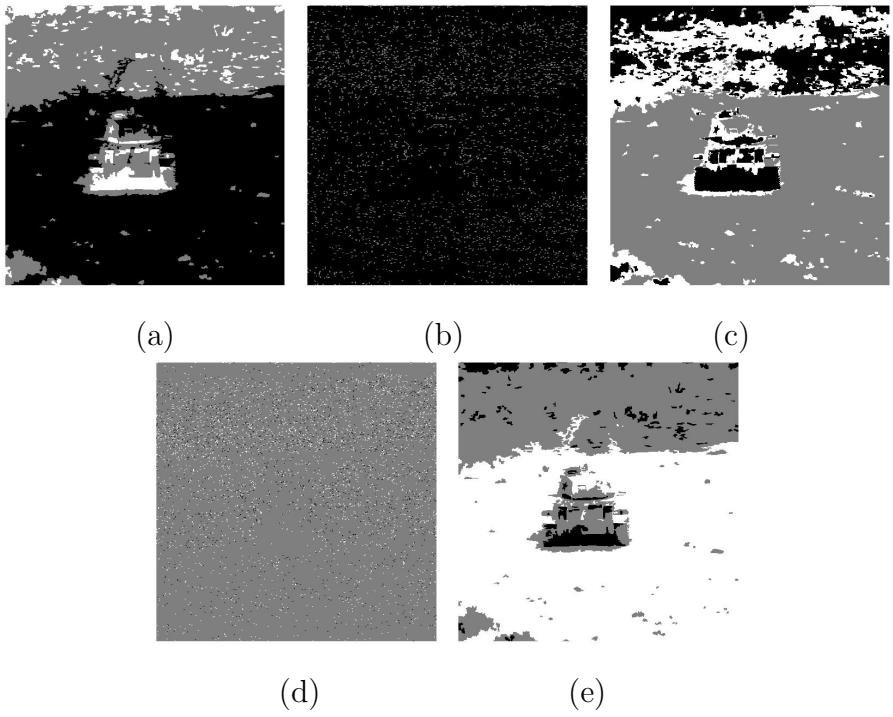


Figure 4.30: *ICM segmentation of the Tank image into three clusters using the DPT and the total variation spectrum shown in Figure 4.29 (a) all 5 TV bands (b) scales 70001 - 120000 (c) scales 120001 - 130139 (d) Scales 1 - 70000 (e) Scales 70001 - 130139*

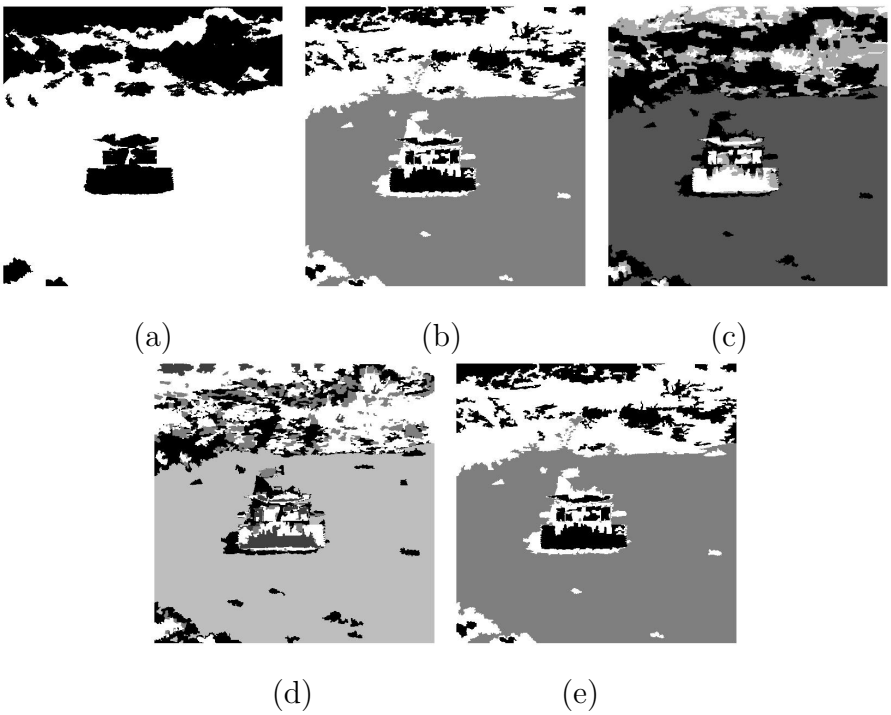


Figure 4.31: *The ICM algorithm illustrated on the Tank image using only pulses of size 100 and larger (a) 2 clusters (b) 3 clusters (c) 4 clusters (d) 5 clusters (e) 3 clusters using the TV bands*

The results presented in this section are by no means the end-all of segmentation with the DPT and have been presented as an indication of the usefulness of the DPT in image segmentation. Future research will involve making use of the LULU scale-space to determine the number of clusters beforehand as Nakamura and Kehtarnavaz [155] do with the Gaussian scale-space; use the scale-space life-times for the segmentation as these may clearly distinguish noise, texture, small detail and large detail; look at alternative connectivity approaches for improved segmentation such as the work done by Soille in [217]; comparisons with state-of-the-art segmentation; and using the shape measures, such as the shape number and shape dispersion matrix, discussed in detail in Section 4.8.2 to obtain further improved segmentation as Urdiales et al [227] do. This last approach ventures into the realm of pattern recognition which will enable the modeling of backgrounds in images and the subsequent removal of them for accurate target detection and tracking.

4.9 Conclusion

In this chapter we have presented an overview of the development of the original Gaussian scale-space of Witkin and Iijima, further works resulting from this, as well the various pre-scale-space notions of introducing scale into analysis of signals and images. We also briefly listed the numerous applications of scale-spaces in image analysis. Most importantly, we provided a formal definition of a scale-space (Section 4.6), which has not been done to our knowledge. The Discrete Pulse Transform results in a scale-space, named the LULU scale-space, according to this definition and we prove this in Section 4.7. The opportunity to investigate the practical use of the LULU scale-space is now available and we delve into this in Sections 4.8.2 and 4.8.3.